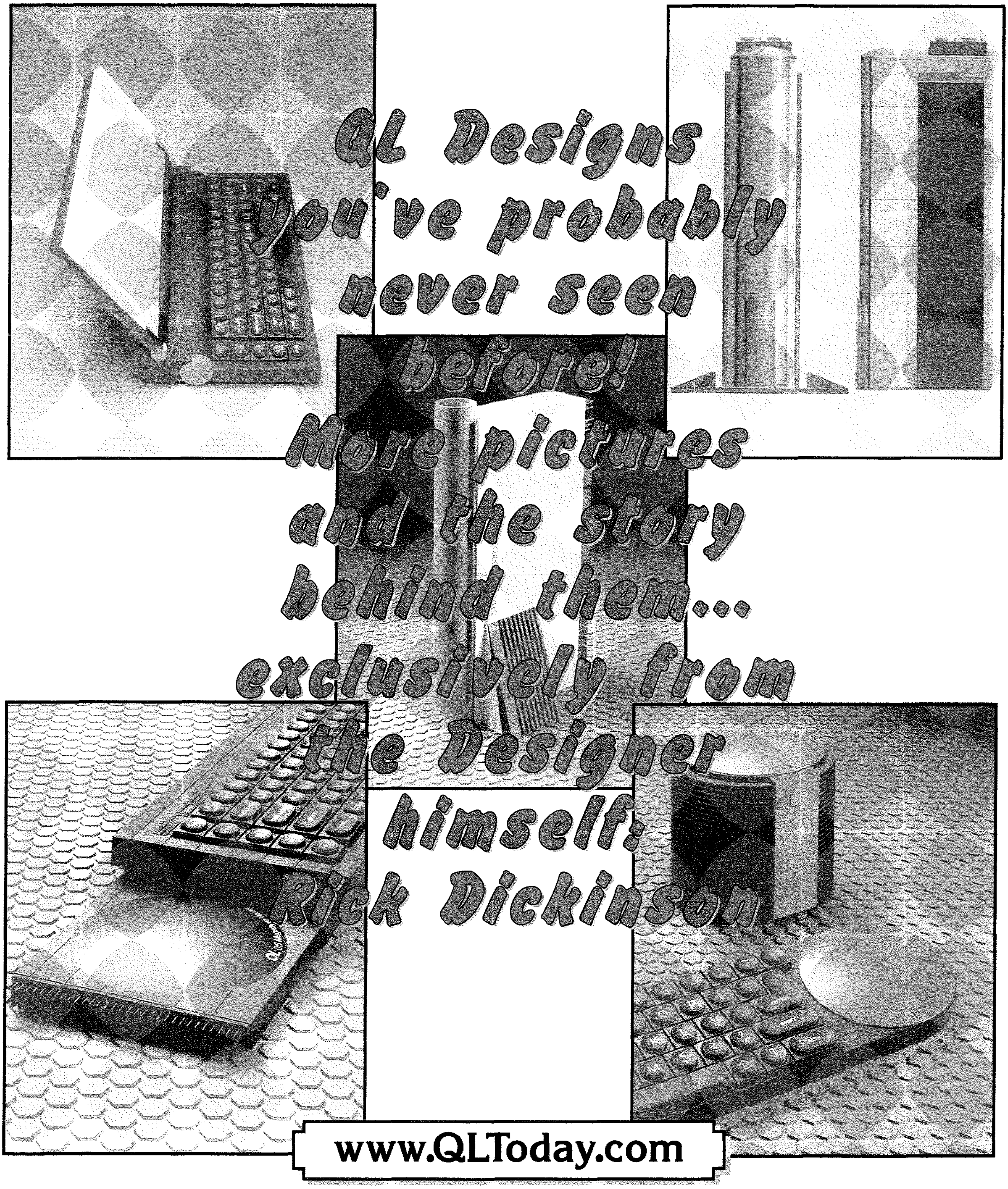
QL Today

Volume 12
Issue 1
Sept.-November
2007

ISSN 1432-5454

The Magazine about QL, QDOS,
Sinclair Computers, SMSQ...



QL Designs you've probably never seen before!

More pictures and the story behind them... exclusively from the Designer himself: Rick Dickinson

www.QLToday.com

Contents

Advertisers

in alphabetical order

QL *Today*

ISSN 1432-5454

German office & Publisher:

Jochen Merz Software
Kaiser-Wilhelm-Str. 302
47169 Duisburg
Germany

Tel. +49 203 502011
Fax +49 203 502012
email: JMerz@j-m-s.com
email: QLToday@j-m-s.com

English office:

Q Branch
20 Locks Hill
Portslade
BN41 2LB
United Kingdom

Tel. +44 1273 386030
Mobile +44 7836 745501
Fax +44 1273 381577
email: qbranch@qbranch.demon.co.uk
email: QLToday@j-m-s.com

Editor:

Geoff Wicks
5b Wordsworth Avenue
Sinfin
Derby DE24 9HQ
United Kingdom

Tel. +44 1332 271366
email: gwicks@beeb.net
email: QLToday@j-m-s.com

Co-Editor:

Bruce Nicholls
38 Derham Gardens
Upminster
Essex RM14 3HA
United Kingdom

Tel +44 20 71930539
Fax +44 870 0568755
email: qltoday@q-v-d.demon.co.uk
email: QLToday@j-m-s.com

QL *Today* is published quarterly, our volume starts middle of September. Please contact the German or English office for current subscription rates or visit our homepage www.QLTODAY.com.

We welcome your comments, suggestions and articles. YOU make **QL *Today*** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email. We prefer ASCII, Quill or text87 format. Pictures may be in _SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

QL *Today* reserves the right to publish or not publish any material submitted. Under no circumstances will **QL *Today*** be held liable for any direct, indirect or consequential damage or loss arising out of the use and/or inability to use any of the material published in **QL *Today***. The opinions expressed herein are those of the authors and are not necessarily those of the publisher.

If you need more information about the UNZIP program which is used by our BOOT program to unpack the files, we suggest that you visit Jonathan Hudsons web site where you find more information about lots of interesting QDOS software and INFOZIP at www.bigfoot.com/~jrhudson/

The deadline for the next issue is the 5th of November 2007

Editorial

by Geoff Wicks

I am tempted to go biblical and enthuse about the alpha and the omega - the beginning and the end.

Our lead article goes back to the early days of the QL. Rick Dickinson, one of the original senior QL designers, describes his experiences as part of the Sinclair team. We are proud of this scoop that is down to Jochen and not the editor. (Although perhaps the true credit should go to Malcolm Cadman who first broke the story on his website.)

Rick Dickinson describes the work he was doing more than two decades ago, but his enthusiasm still shines through his article.

That was the alpha, but what of the omega?

This is the third consecutive issue that Quanta has provided our lead news story. The first two were positive stories describing the fruits of the present committee's efforts to improve the organisation.

We may not always agree on policy and tactics, but few would deny the present Quanta officers are hard working. Quanta may have lost numerous members, but in other ways it is in a better shape than it has been for some time. There is now more openness on committee meetings; a magazine restored to its former glory; a greater willingness to make constructive use of its capital; and a renewed website.

This time our Quanta story is a sober one. Even more sobering is the fact that our source material is the Quanta Magazine and a senior member of the Quanta committee.

Quanta is rapidly becoming the QL's equivalent of the Church of England. An established institution that is seen as an essential part of the community, but which few people actually use. About four-fifths of Quanta members are totally passive. They do nothing more than pay their subscription. Even the active one-fifth show a marked reluctance to play their part in running the organisation by serving on the committee.

No committee means no Quanta. Use it or lose it.

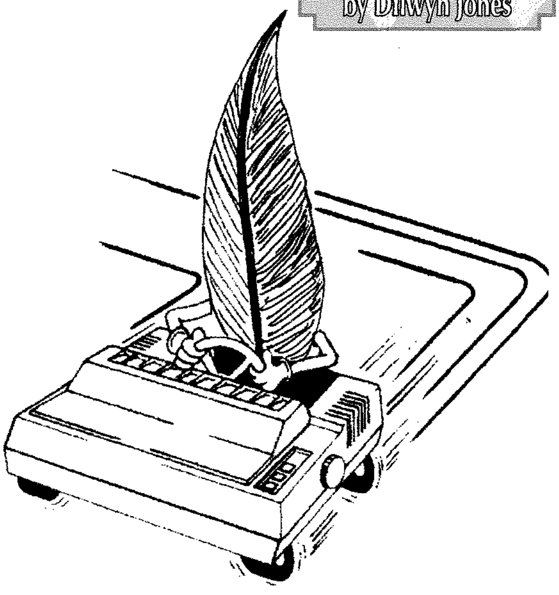
Both Rick Dickinson and the present Quanta Officers have set an example to us by the enthusiasm with which they have done their jobs. There is still time for the rest of us to delay the omega for a good few years yet, but we will need to recreate the enthusiasm of the early days.

Quanta's committee are planning a party to celebrate the QL's quarter centenary in 2009. It will be a good opportunity for the rest of us to show our appreciation of their work and achievements. What better way to do this than for others to take over and continue to build on their work with the same enthusiasm?

We are the people who will finally decide if the big Quanta event of 2009 is to be a party or a wake.

Cartoon

by Dilwyn Jones



Quill Printer Driver?

QUANTA Closure imminent?

A senior member of the Quanta committee has given a stark warning that Quanta could be forced to close down in just over 18 months time. Writing in the June/July 2007 Quanta Magazine **John Gilpin** names two factors threatening Quanta's future viability. One is the 2005 changes to the constitution that require some of the present officers to step down from the committee in 2009. The other is that Quanta members are reluctant to serve on the committee. If Quanta cannot replace the committee members stepping down, it will be forced to wind up early in 2009. Its capital would then be donated to the British Red Cross Society and a five figure sum lost to the QL community.

About four out of five Quanta members play no part in the life of the organisation other than to pay their subscription, but even the active members are not willing to join the committee. John Gilpin points out that in 2000 the committee shared ten specific jobs among seven members. Now they are down to four members and those ten tasks still have to be shared among them. John himself trebles up as Treasurer, Membership Secretary and Acting Joint Magazine Editor. He adds that since 2000 only 13 people out of a current membership of about 200 have been prepared to serve on the committee.

YOUR QUANTA NEEDS YOU **John Gilpin**

Over the last seven years, there have been only thirteen different people involved on the Quanta Committee. At the beginning of this century, the committee was seven strong carrying out ten specific jobs. Anyone who has held an honorary position on a committee will accept that it is difficult to find people (with the required skills) who will give up a little of their time to serve any organisation.

QUANTA whilst still carrying out the same ten specific jobs now has only four committee members who, between them, are attempting to keep their heads above water and satisfy a membership of almost 200 members. While some of the double jobs we do go nicely with one another, there are other jobs which are at risk of not being done at all unless we can involve some new people to help.

Changes to the Quanta constitution in 2005 imposed restrictions on the length of time committee members can remain on the committee and this applies to some of the present officers in 2009. **John Gilpin** writes:

"Personal requests for other members to replace them have fallen on deaf ears so this is our appeal to all the remaining QUANTA members to get involved NOW or is it the request of the membership that QUANTA should be wound up in the next couple of years and give all our assets to the British Red Cross Society as detailed in the constitution?"

The 2005 constitutional changes were the brainchild of chairman John Mason and were steamrollered through Quanta with the minimum of consultation and discussion. However they received almost unanimous support from the then committee and were approved overwhelmingly by those members who used their voting rights.

John Mason sprang the constitutional amendments on the committee at a meeting held at Bicester Cherwell Valley on Sunday 6th February 2005. Following this meeting one committee member emailed the rest of the committee disputing whether the committee approved of the changes: *"Several members of the committee, including myself, expressed doubts over these amendments. Although I can see some merit in the changes, I personally expressed the opinion that they had been sprung on us without warning and needed more consideration of the likely practical effect on Quanta. No vote was taken on this issue and I had the impression that the consensus of the committee were for the amendments to lie on the table for further consideration."*

Following this email John Mason emailed all members of the committee inviting further comments and objections. No member of the committee replied thus satisfying the dissenting member that all the other committee members were in full agreement with the proposed amendments.

Quanta members were also given little formal opportunity to debate the amendments, but a fierce discussion took place on the QL-users email group. Although there was some vigorous opposition to the proposed amendments, in subsequent voting members approved the changes by a clear majority. There were 39 votes for, 13 against and 4 abstentions.

The intention behind the changes was to bring freshness and continuity to the Quanta committee, but paradoxically Quanta is now in danger of being strangled by its own constitution. An important aspect of the changes was to avoid a situation where more than one officer would step down in any one year, but some inherent contradictions in their wording combined with the lack of phased implementation have led to just that situation arising.

Although Quanta will almost certainly still have a committee until Spring 2009, the crisis is an immediate one. Several committee tasks require specific skills such as John Gilpin's work as treasurer. Both John Gilpin and John Mason, who have both been key figures in restoring the fortunes of the Quanta Magazine, are due to stand down. They will be unable to continue as joint acting editors of the magazine after Spring 2009 as this is a committee post. For Quanta to survive "officers in waiting" need to be gaining committee experience from next year's AGM.

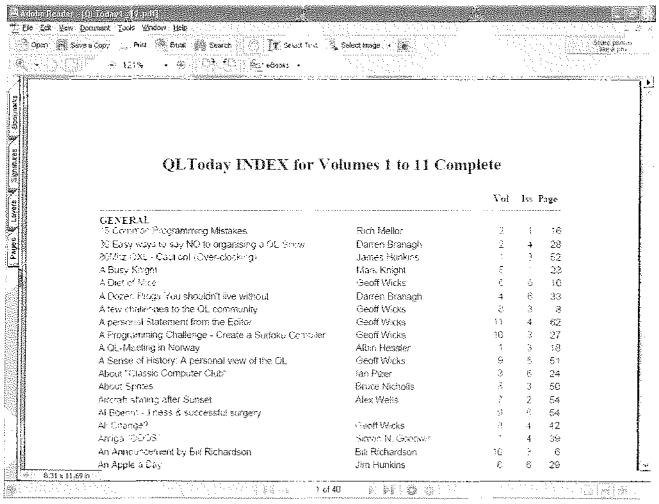
QL Today Index

Brian Kemmett has released an index to volume 11 of QL Today. He has also made improvements to the indexes of previous volumes. **Dilwyn Jones** writes:

"Brian Kemmett has now compiled an index to the current Volume 11 of QL Today. It is available in a variety of formats from Dilwyn Jones's website.

There is a single PDF file listing all 11 volumes of the magazine, consisting of a single 40 page index. The individual volume indexes are available in plain text, Quill DOC and word DOC files as well.

Being text, all can be searched in their respective viewers or word processors, including the PDF version, useful when you need to find that article you need!"


Adobe Reader - [QLToday1_11.pdf]
File Edit View Document Tools Window Help
Open Save a Copy Print Email Search Select Text Select Image
121%
eBooks
Bookmarks
Signatures
Layers
Pages
8.31 x 11.69 in
1 of 40

QLToday INDEX for Volumes 1 to 11 Complete

		Vol	Iss	Page
GENERAL				
15 Common Programming Mistakes	Rich Mellor	2	1	16
20 Easy ways to say NO to organising a QL Show	Darren Branagh	2	4	28
20Mhz QXL - Caution! (Over-clocking)	James Hunkins	1	3	52
A Busy Knight	Mark Knight	5		23
A Diet of Mice	Geoff Wicks	6	6	10
A Dozen Progs You shouldn't live without	Darren Branagh	4	6	33
A few challenges to the QL community	Geoff Wicks	8	3	8
A personal Statement from the Editor	Geoff Wicks	11	4	62
A Programming Challenge - Create a Sudoku Compiler	Geoff Wicks	10	3	27
A QL-Meeting in Norway	Albin Hessler	1	3	18
A Sense of History: A personal view of the QL	Geoff Wicks	9	5	51
About "Classic Computer Club"	Ian Pizer	3	6	24
About Sprites	Bruce Nicholls	8	3	50
Aircraft shining after Sunset	Alex Wells	7	2	54
Al Boehm - Illness & successful surgery		9	4	54
All Change?	Geoff Wicks	8	4	42
Amiga QDOS	Simon N. Goodwin	1	4	39
An Announcement by Bill Richardson	Bill Richardson	10	2	6
An Apple a Day	Jim Hunkins	6	6	29

The QL Today team are grateful to Brian Kemmett for his work in indexing QL Today. The editor makes extensive use of the index when researching stories or checking information. He can highly recommend the new pdf index of all 11 volumes, which has fast search facilities.

The index can be downloaded from the QL Today page on Dilwyn's website:

www.dilwyn.uk6.net/gen/qltoday/qltoday.html

QL Today Writers

A QL Today reader has suggested we publish a list of contact email addresses of our writers. We contacted almost all the people who have written for us during the last two volumes asking their opinion and had a good response. Most of our regular writers are happy to have a contact email address published and the list appears elsewhere in this issue.

Please note that our writers are all busy people who may not be able to respond immediately. We would ask you to use this facility responsibly.

A few of our writers indicated they preferred not to have their email addresses published. There are many reasons for this including problems with spam or personal circumstances that make it difficult to receive and respond reliably to emails. In most cases QL Today will contact a writer on your behalf.

Early Sinclair Designs

Malcolm Cadman has posted some early Sinclair design information on his website. He writes:

"I have added some links to my web site of the emerging Sinclair Industrial Design details that are being put up by Rick Dickinson - former Senior Designer at Sinclair Research.

The first link is to Spectrum designs, the second to a possible future successor to the QL, and the third to the more recent Gizmondo designs.

More detailed information of the QL design is still to be forthcoming.

I have also added a link to the QL Wiki by Rich Mellor.

All at:

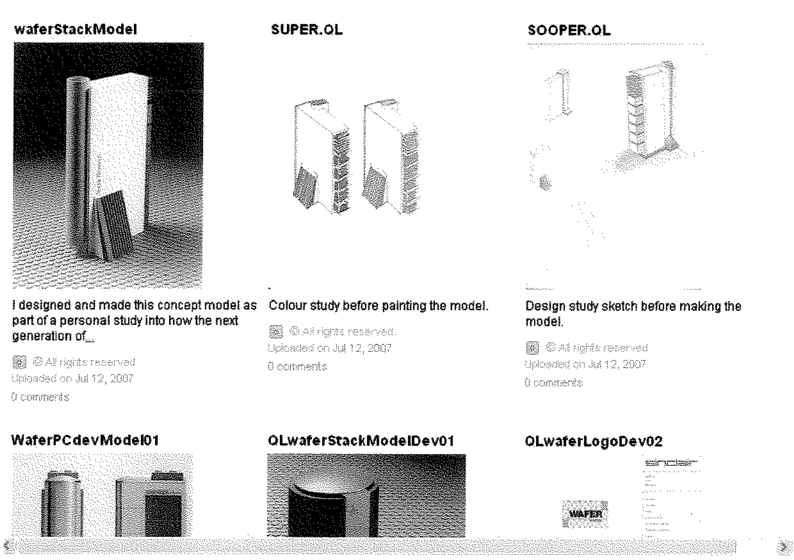
http://www.mcad.demon.co.uk/lquan.htm

The new links are to be found near the bottom of the page.

Check it out, when you can.

With a dial-up account there are a lot of photos to load and view, so expect a reasonable time to download. Then save the web pages locally on a hard drive, or similar. Once they are in the cache of the web browser, they are easier to view the next time.

With a broad band account, or similar there will be no problems."



waferStackModel

I designed and made this concept model as part of a personal study into how the next generation of...

© All rights reserved
Uploaded on Jul 12, 2007
0 comments

SUPER.QL

Colour study before painting the model.

© All rights reserved.
Uploaded on Jul 12, 2007
0 comments

SOOPER.QL

Design study sketch before making the model.

© All rights reserved
Uploaded on Jul 12, 2007
0 comments

WaferPCdevModel01

QLwaferStackModelDev01

QLwaferLogoDev02

He then adds by way of explanation:

"The designs are a record of the "retro-history" of Sinclair designs and ideas.

They show how products that did reach the market - like the Spectrum (ZX82), and the Sinclair QL (ZX83) - were developed as concepts and ideas.

Other ideas shown were in progress of potential development in the 1980's and 1990's, yet did not come about with the demise of Sinclair Research.

It also shows that it is not only Apple Computers that have the "wow factor" in terms of product appeal and functionality.

The concept ideas for a "ZX84" - a successor to the QL - as a tower case with a small desktop footprint still looks modern today."

Elsewhere in this issue you will find an article by Rick Dickenson.

The Dilwyn Jones Spot

As ever Dilwyn helps us to fill our news pages with his prolific programming output. This time he has two new programs and some upgrades:

ZIP MANAGER

Dilwyn Jones writes:

"I've added a new program called Zip Manager to my website. This freeware program acts as a pointer driven front end for the Zip and Unzip programs, to make them a bit easier to use.

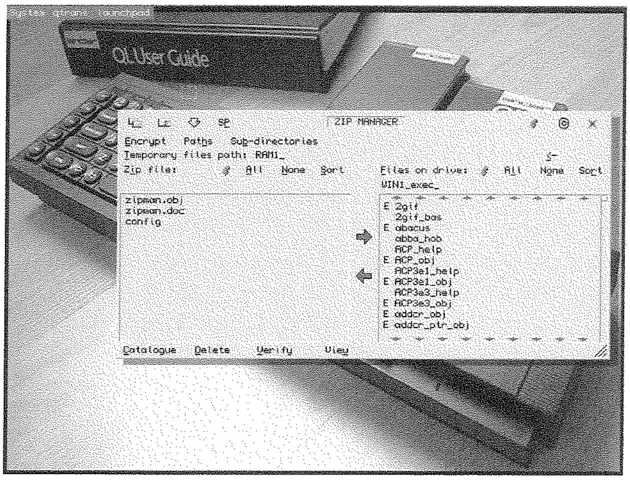
Written as a follow up to the Zip and Unzip short series in QL Today, this program is a menu driven front end system, based on my Q-Trans program. It requires Window Manager 2 (uses System Palette colour themes) which means you must have SMSQ/E version 3 or later, or QDOS with pointer environment version 2 or later.

You can add or delete files to or from a selected zip archive, and delete or view files. If you use FileInfo 2, Zip Manager is aware of that and so viewing may be achieved via the file associations defined. If you have a favourite editor or viewer program such as QD or S-Edit, you can set up Zip Manager to fire up that program to view files contained in zip files.

Zip Manager lets you choose whether filename paths or just pure filenames are stored, lets you add password protection to new Zip files if required, and even control whether Zip processes sub-directories when adding to zip files.

Zip Manager is available to download from my website at:

http://www.dilwyn.uk6.net/arch/index.html



```
Temporary files path: RAM1_
Zip file:                          Files on drive:
                                   WIN1_exec_
zipman.obj                         E 2gif
zipman.doc                           2gif_bas
config                             E abacus
                                     abba_hob
                                     ACP_help
                                   E ACP_obj
                                     ACP3e1_help
                                   E ACP3e1_obj
                                     ACP3e3_help
                                   E ACP3e3_obj
                                   E addcr_obj
                                   E addcr_ptr_obj
```

GO!

Dilwyn writes:

"Go! is a new program from Dilwyn Jones. It is yet another of those program launching programs, but this time not a GUI or anything too graphical or fancy. More like a Start menu in some ways. It's based on the Quicklaunch menu from Launchpad.

Basically, teach it what program to start (filename, job name, program name and any special DEV, PROG_USE, DATA_USE settings) and it will store those settings, letting you just click on that program's name in the future to start the program.

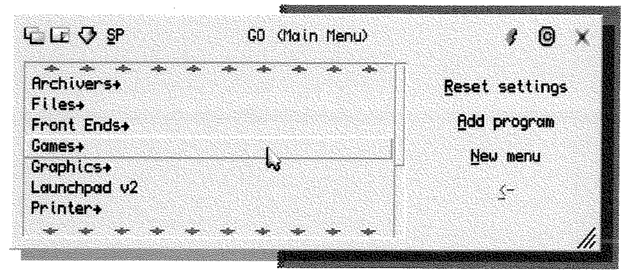
You can build a set of menus and sub-menus and add programs to them as required. It can store details for hundreds of programs if need be.

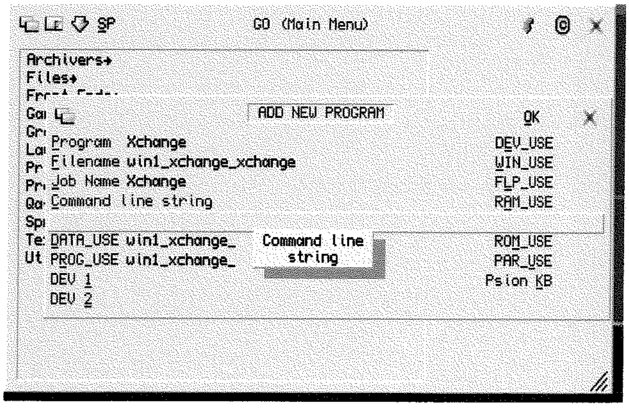
Go! is pointer driven and needs Window Manager 2 (which means SMSQ/E version 3 or later, or QDOS with pointer environment version 2). It will also need expanded memory and Toolkit 2.

Above all, Go! is meant to be a simple to use program starting system for those who do not want the complexities of a Graphical User Interface (GUI) like QDT or Launchpad.

Go! may be downloaded from my website (a 60KB download) at

www.dilwyn.uk6.net/utils/index.html





```
Program  Xchange
Filename win1_xchange_xchange
Job Name Xchange
Command line string

DATA_USE win1_xchange_
PROG_USE win1_xchange_
DEV 1
DEV 2
```

Cocktails Waiter

James Bond is shaken, if not stirred by this message from **Dilwyn**:

"The formerly commercial program Cocktails Waiter has now been released as freeware and is available from the Miscellaneous Programs page on my website.

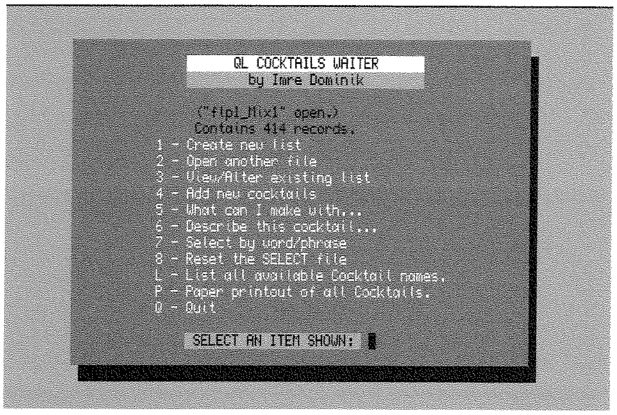
Cocktails Waiter contains 3 databases totalling about 1,000 cocktail drink recipes. You can search, print, view and select cocktail recipes by ingredients or by names. Although not specifically intended as such, it does have facilities to set up new databases so could probably be used for cooking recipes as well.

Unfortunately, I do not have access to the source files for this program, but the databases are standard Archive databases so could be exported for use with other QL database programs such as Data Design or Easybase. I have also updated the manual.

This program is NOT an excuse to turn up anything less than sober for a Quanta AGM, of course!

The program can be downloaded from:

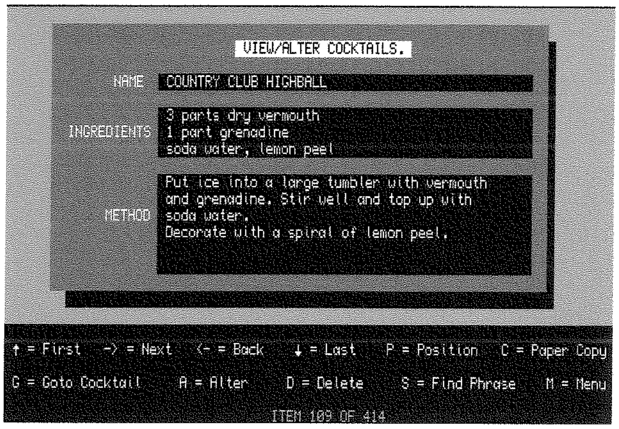
http://www.dilwyn.uk6.net/misc/index.html



```
              QL COCKTAILS WAITER
                by Imre Dominik

              ("flp1_Mix1" open.)
              Contains 414 records.
          1 - Create new list
          2 - Open another file
          3 - View/Alter existing list
          4 - Add new cocktails
          5 - What can I make with...
          6 - Describe this cocktail...
          7 - Select by word/phrase
          8 - Reset the SELECT file
          L - List all available Cocktail names.
          P - Paper printout of all Cocktails.
          Q - Quit

             SELECT AN ITEM SHOWN:
```



```
                     VIEW/ALTER COCKTAILS.

         NAME  COUNTRY CLUB HIGHBALL

               3 parts dry vermouth
  INGREDIENTS  1 part grenadine
               soda water, lemon peel

               Put ice into a large tumbler with vermouth
               and grenadine. Stir well and top up with
       METHOD  soda water.
               Decorate with a spiral of lemon peel.

↑ = First   -> = Next   <- = Back   ↓ = Last   P = Position   C = Paper Copy

G = Goto Cocktail   A = Alter   D = Delete   S = Find Phrase   M = Menu

                         ITEM 109 OF 414
```

PCB CAD Update

Dilwyn also has news of a further update to PCB CAD:

*"**Malcolm Lear** has kindly sent me v6.16 of his PCB Cad program, which I have uploaded to the Graphics page on my website. Here is Malcolm's description of the update:*

'Quite a few changes this time. Most important is proper behavior on shutting down and releasing allocated memory and a persistent bug that stomped on the PE has been sorted.'

It may be downloaded from:

http://www.dilwyn.uk6.net/graphics/index.html

General DJ Software Updates

Dilwyn Jones writes:

"I have placed three updates to my programs on my website:

1. Launchpad 2.07 demo version and full version update (for registered users via password protected zip file). To download, go to the Launchpad page on my website and follow the links to the downloads page. V2.07 fixes a small problem in the program window resize routines, *allowing the resized display to appear at a more logical position than the semi-random positions generated before, and updates the Easyptr extensions to latest versions, and a few code tidy-ups.*

2. Q-Trans v2.06 also fixes the same problem in the resize routines and also updates the Easyptr extensions used. It may be downloaded from the same page as Launchpad above. Note that Q-Trans itself is freeware, you do not have to be a registered user to be able to use Q-Trans as a stand alone file handling program.

3. Zip Manager v1.02 fixes a bug whereby the program would not run on a 512x256 QL display because of a problem with program shadow size, and also fixes a potentially serious bug in the configuration block, along with introducing a couple of new items in the configuration set-up. This program update can be downloaded as a zip file from the Archivers page on my website. Zip Manager is a freeware program.

As usual with my recent programs, all require the use of Window Manager 2, which basically means you should be running SMSQ/E v3.00 or later, or QDOS with pointer environment v2.00 or later.

The updates can be downloaded from their respective pages on my website, at:

http://www.dilwyn.uk6.net/index.html

SUQCESS Upgrade

Bob Spelten writes:

"Suqcess continues with version 2.05.

The upgrade as presented in Eindhoven still had some bugs but I hope I have fixed them all by now.

The biggest change is in the ability to use subfields. Text fields can be divided into subfields by a linefeed character. Fields up to 4000 characters can now be used. This had major implications for the Export, Import, Edit, Scrap and Print routines.

Export will convert the linefeed character to a vertical line so that other Import programs can safely be used. Although exporting for Psion will limit fields to 253 characters.

Import can now handle fields larger then 128 characters but the lengths must be input for each text field.

Text Editing can be done as one long string or as an array with each subfield on a separate line. Texts can be split or joined into subfields.

The Edit window can also be used to just view long fields with editing disabled.

Copy-to-File with TAB's between the fields, can now also include the field names.

Printing can be done in columns as in the main display or in rows like the View window and with record numbers. Then the field names can also be printed.

A DO on a not selected field, when some records are selected, will now let you edit this field and not open the View window. Only a DO on a selected record will open the View window.

When you are on the last of a group of selected records in the View window and you hit the <+> line, "End of selection" now lets you go back to the last viewed record.

In the Find menu the Case On/Off option can actually be used.

The Readme, Changes and Help files that come with the program hold more details on improvements and bug fixes.

As always, full working copies (in English, German or Dutch) can be obtained from Jochen Merz Software or Q Branch.

The English trail version on Wolfgang Uhlig's site www.uhlich.nl/ql is still the 2.04 version. My site is still under construction but the new 2.05 can be downloaded through there:

http://members.upc.nl/b.spelten/ql

This is actually a link to the Dilwyn Jones site:

http://www.dilwyn.uk6.net/demos/index.html

Party Time?

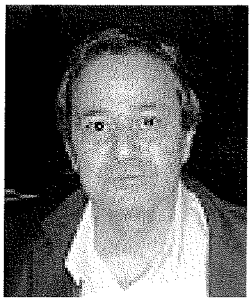
Quanta has announced its intention to organise some form of celebration of the QL's quarter centenary in 2009. Secretary ***Sarah Gilpin*** refers to "a party", although chairman John Mason suggests a "QL is 21" type event. Whatever the form of celebration it will be open to non-Quanta members.

Quanta is asking for ideas, suggestions, thoughts and volunteers. Please email these to:

secretary@quanta.org.uk

Paul Merdinian

Quanta has informed us of the death of Paul Merdinian on 21st July 2007.



Paul was at times a controversial person within the QL community, but was best known as an obsessive collector of word lists. He was the author of RWAP's half a million word P-Word English dictionary.

In the early 1990's he provided accurate wiring details for QL RGB to SCART cables, something that proved rather more complicated than it appeared at first sight.

QUANTA's Big Secret

QL sleuth **Per Witte** has exposed one of Quanta's best kept secrets, and his evidence comes from no less a source than the BBC. He writes: *"Nice to see that 'The $100 One Laptop Per Child laptop XO will be produced in Taiwan by Quanta' / see BBC news story* **http://news.bbc.co.uk/2/hi/technology/6908946.stm** *But why have they kept it under their hat(s?), and why did they procure a new, non-QDOS OS for the project?"*

According to the BBC "Quanta is the world's largest laptop manufacturer." And the editor thought he was joking in the last issue when he accused Quanta members of lighting their cigars with £20 notes.

Apology

The editor apologises to **Dan Abbott**, Quanta's webmaster. His name was incorrectly given as Don Abbott throughout the last issue of QL Today.

QL Designs and more... The Cover Story

by Rick Dickinson

This account is taken from the period after the QL launch.

Clive Sinclair had developed a new building outside of Cambridge to house the growing Sinclair Research Ltd (SRL) - called 'Metalab'. Originally Matalab was to be a lab only for a small group of top thinkers called 'Blue Sky'.

At Metalab we first designed the Spectrum+ and 128. It was decided that these two products should have 'conventional' keyboards, and therefore use the same design of keys as in the QL. This made the Industrial Design easy because we simply adopted the QL aesthetic, and used the same manufacturer. One could argue that the Spectrum at this point lost it's individuality - it was not a particularly natural evolution, but typical commercially. The automotive industry does this and occasionally creates a work of art that people desire purely for the aesthetic, and then completely change it at the next generation redesign for something that just looks like all the other cars in their range - 70's Ford Mustang, New Audi TT. How I would love to do the next Spectrum!

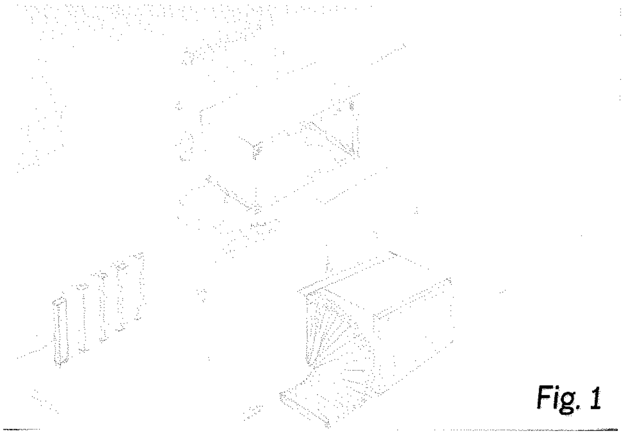


Fig. 1

We started to work on QL expansion by linking an expansion 'hanger' via a flexible cable into the QL's main expansion port. This hanger would take a wide range of PCB's based on the standard Euro size card of that time. Some models were made but only this sketch remains - please see figure 1.

These 'Euro Card' modules could also be plugged individually into the main QL expansion port - but only one at a time. I have two remaining 512 MB Ram cards for the QL, one uses standard DRAM, the other uses a 100mm diameter wafer, see figures 2, 3, 4, 5 and 6:

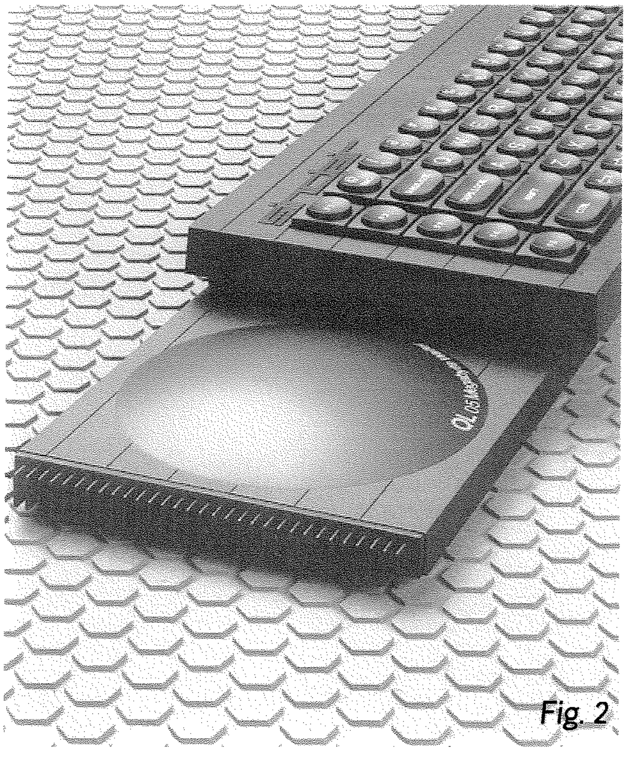
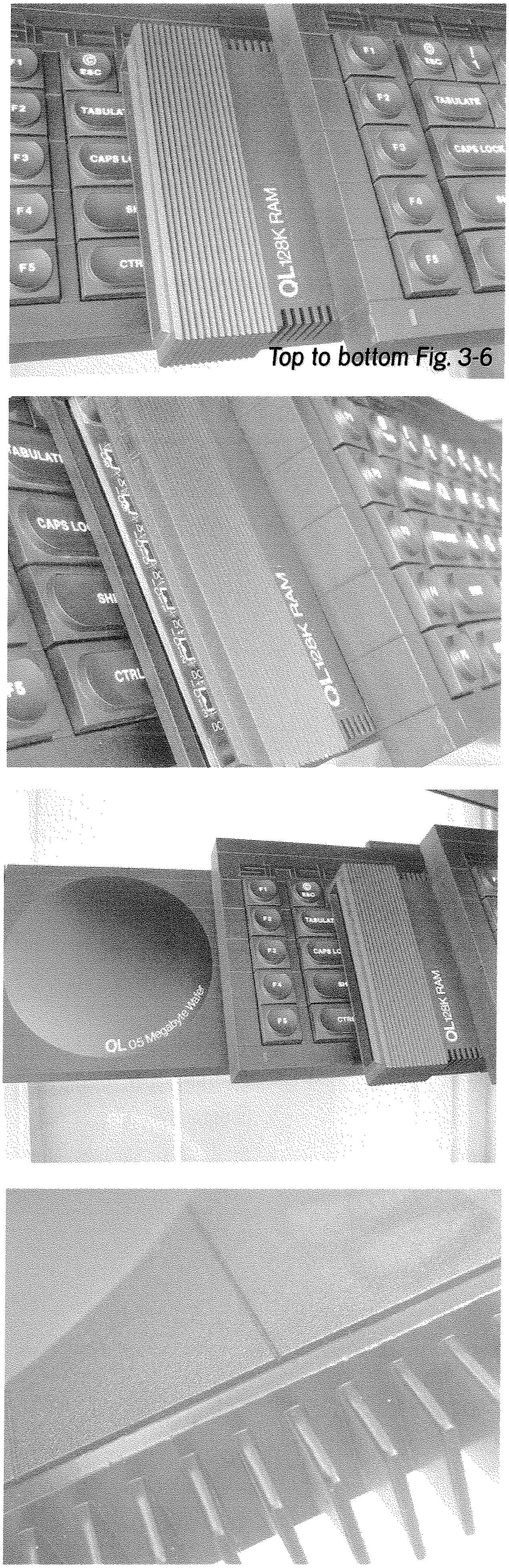


Fig. 2



Top to bottom Fig. 3-6

The wafer naturally was the most interesting from every perspective. It seemed to make sense, theoretically, to keep all the IC's on their original substrate and achieving the highest possible density. The dome on the Wafer moulding is intended to be a semantic that suggests something different and circular within! To get the heat out of the wafer I had to bond it to a PCB carrying thick copper planes and through-hole links which in turn was bonded to a zinc die cast heat sink which I led to the outside of the product into a series of heat dissipating fins. It was a struggle to get the cosmetics up on a manufacturing process that produces parts that are normally hidden. The QL wafer module also carried a battery back up. This was the flat Polaroid battery as used in the TV80. This kept the overall package flat and neat, and I designed a small slider to eject the battery.

All this business with wafers caused a lot of excitement, Clive was in discussions with Ivor Catt, and there was a sense of a huge emerging technological breakthrough. I made a few designs and models – purely at a conceptual level, wondering how we might manage the new devices in product design terms. No one had designed a 'home' computer yet that stood up on end – this would save precious desk space and seemed logical, assuming I could see no adverse reasons elsewhere – so was born the wafer tower, although it could use standard PCB technology too. Please see figures 7, 8, 9 and 10:

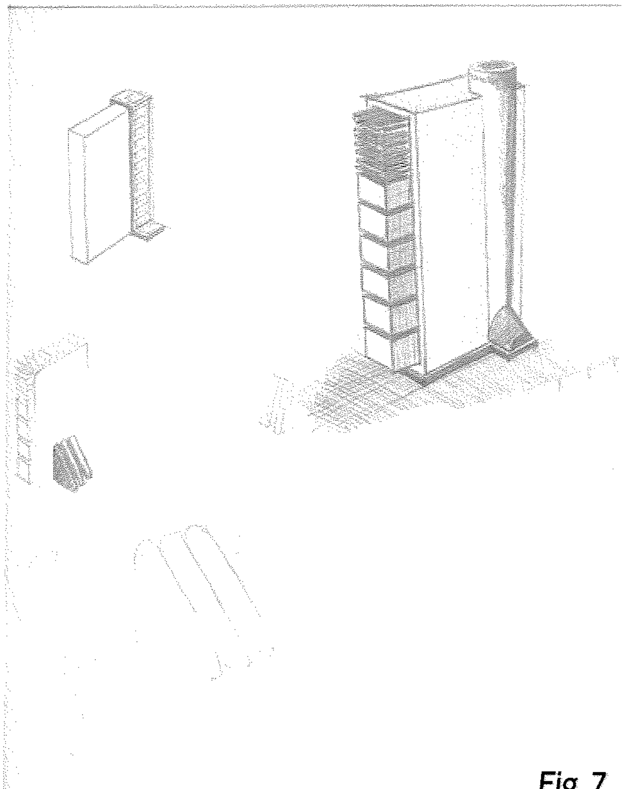
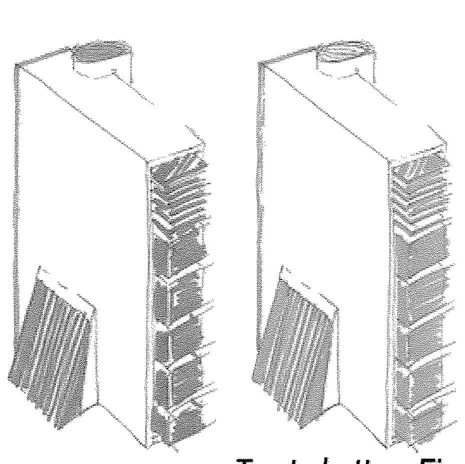
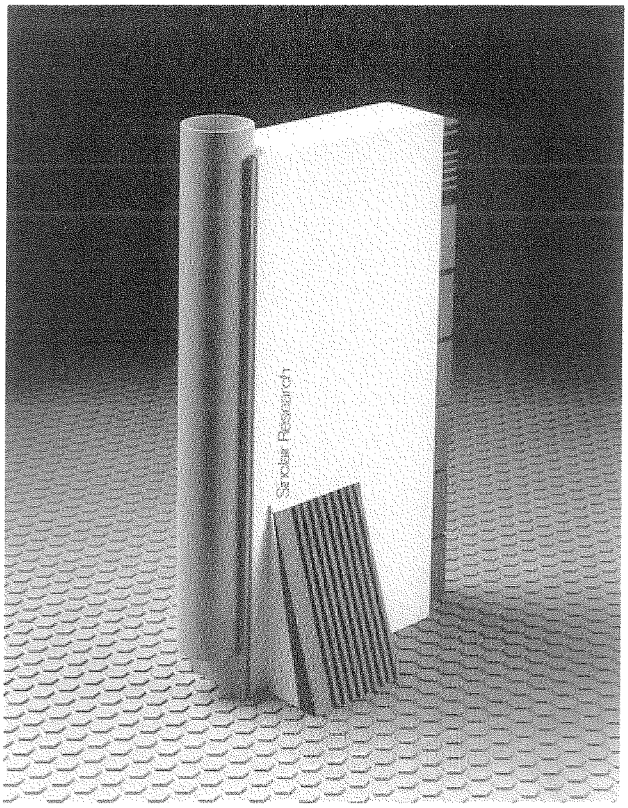
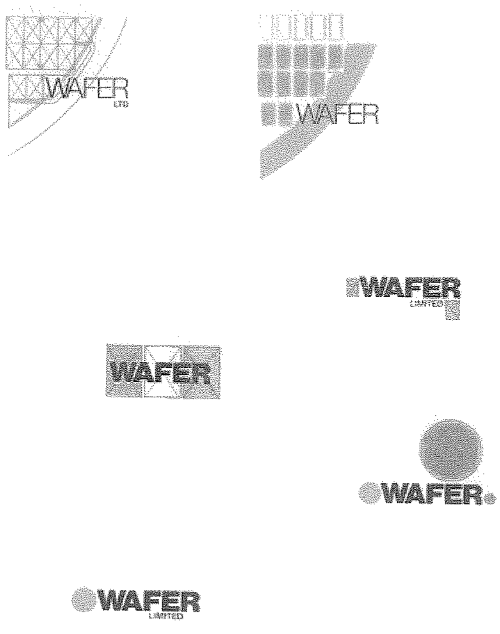


Fig. 7



Top to bottom Fig. 8-10





The connectors would be in a line along the back as in a normal computer, but I designed a half-tube (grey in the image) that could slide over these and collect all the cables like a conduit, and direct them out of the base and away from the desk top. The tube also acted as a chimney to ventilate the case. The black blocks and fins at the front edge are heat sinks from the internal hot spots when using wafers. The triangular block at the base is the power supply, which adds weight and footprint width to improve physical stability.

I then wondered how the next generation QL might be – a minimum size keyboard chassis, with remote processing in a stand-alone wireless box – full of wafers, figures 11 and 12.

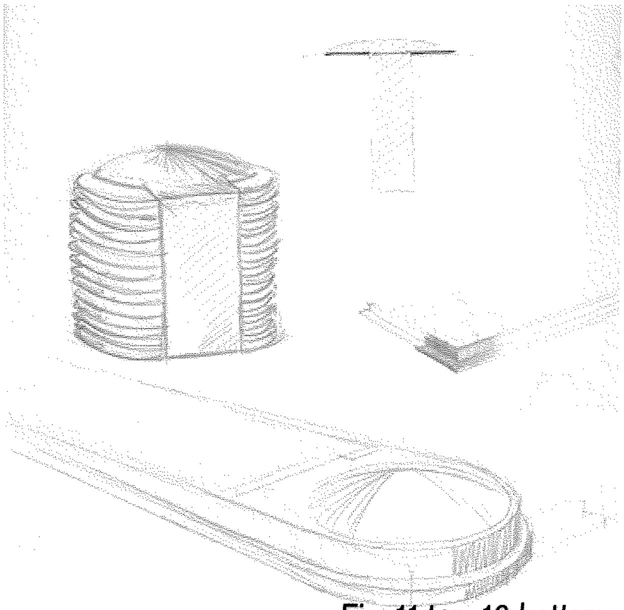
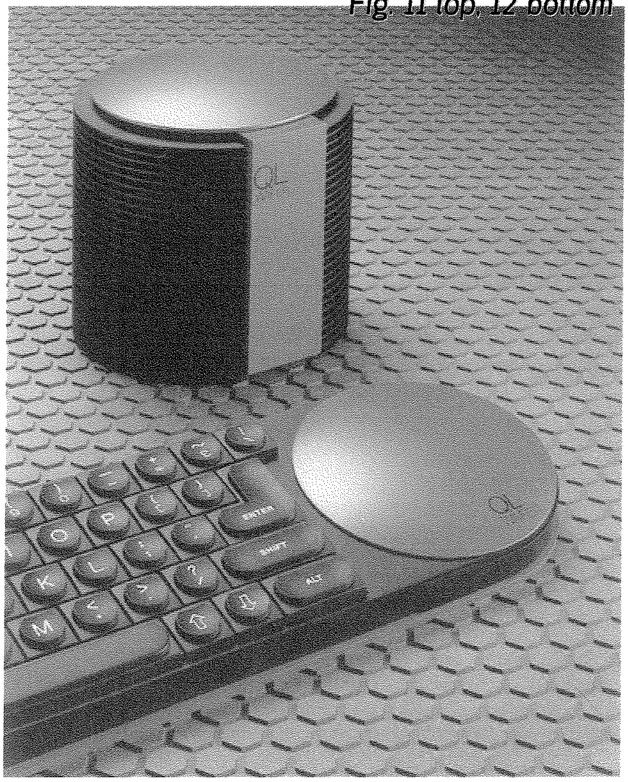


Fig. 11 top, 12 bottom



Clive, I think, had been involved with starting a spin-off called Anamartic to develop wafers, or wafer scale as it was sometimes called, or VLSI (very large scale integration). I produced this futuristic design of how this might go. This was a tower that stood on the floor, 1 meter high. Although the development wafers were to go into production in a more conventional and invisible way by being boxed and fitting into existing manufacturers chassis and cabinets – I wanted to make the technology more 'visible', figure 13:

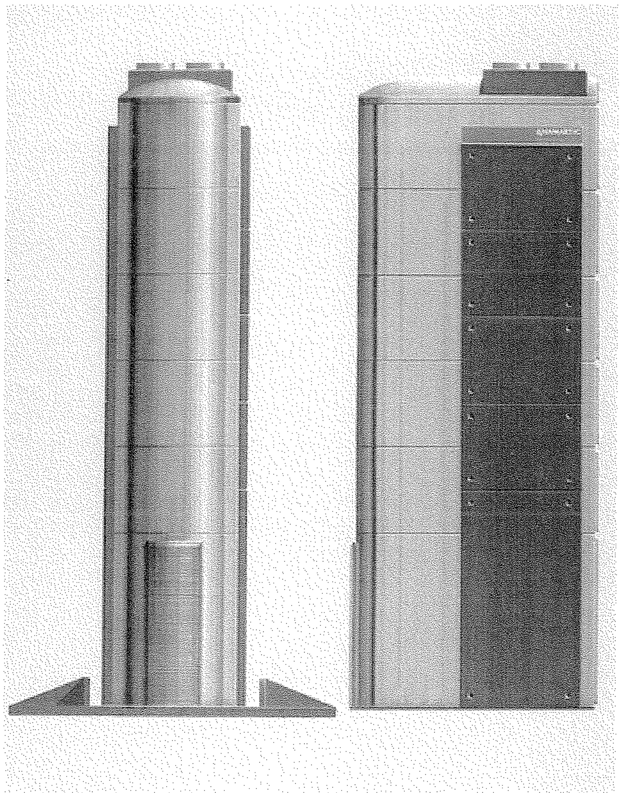


Fig. 13

Whilst all this was going on, two other projects were developing in parallel, one very visible – Pandora, and another very invisible – Loki.
Pandora was to be a Lap top – it's that simple. The difficult part was Clive's insistence on using the TV80 flat cathode ray tube. The inventor Bill Den was tasked with this job – he and Clive would spend the early hours of every morning at Metalab going over ideas and calculations, then Bill would design and manufacture an optical system in the work shops, I can see him knee deep in Acrylic swarf at the milling machine. The results would be analysed, and then the whole process would be repeated with improvements, week after week after week. Sadly the results either made the image look as though it was buried in the floor below, or somewhere between your ears –it was brilliant work but too much for the tube.

Many working prototypes were built, and some static models. The first here uses a double hinge – the cover lifts to expose the keyboard only, and then all of the top hinges back to reveal the display, figure 14:

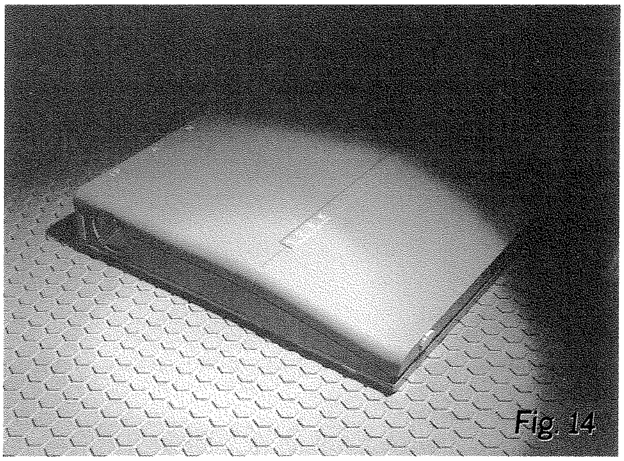


Fig. 14

The next one, figure 15, has a single large cover as a protective lid only.

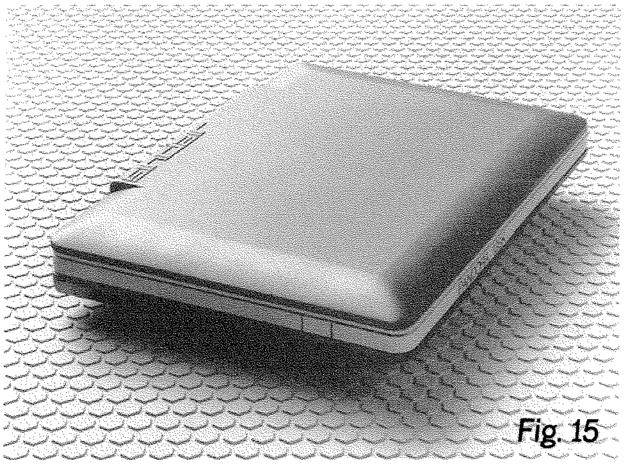


Fig. 15

Eventually I did this one using an LCD – I guess I nearly got fired at that point?! (Figure 16 and 17)

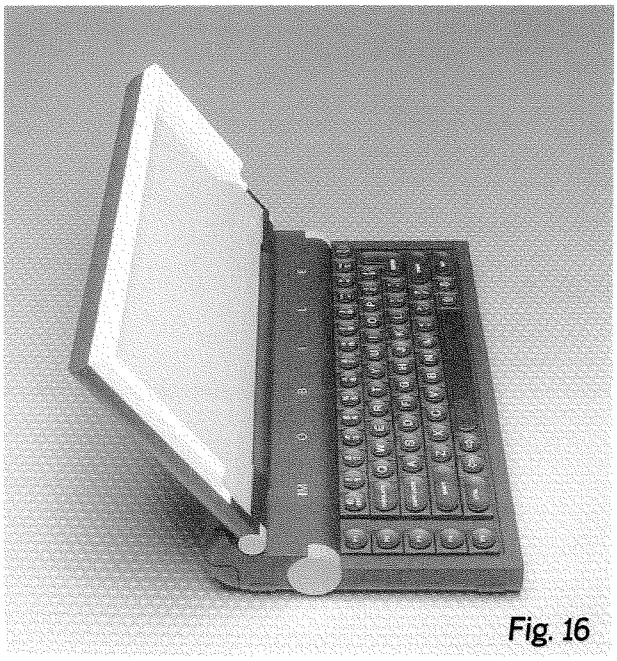


Fig. 16

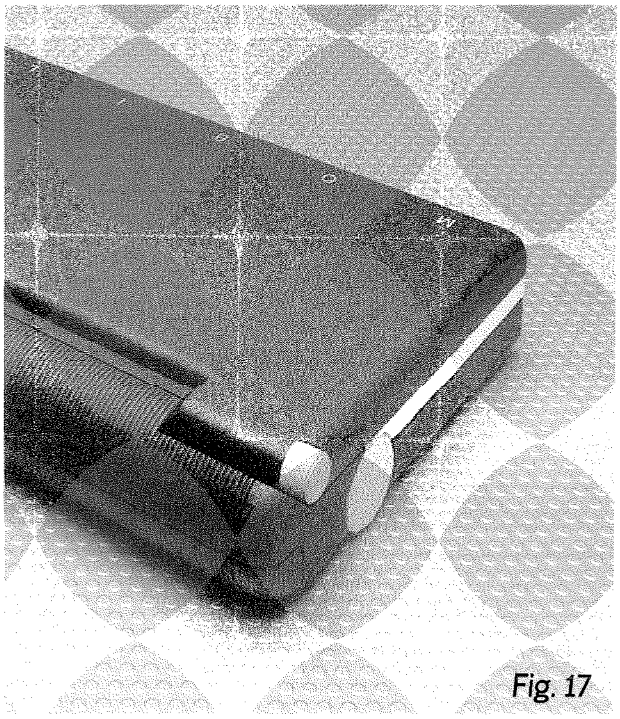


Fig. 17

Pandora also had micro drives - we didn't want to use those either - Pandora was a box to contain all Sinclair Technology, we would have rather used LCD's and Floppy Drives and so on.
Colour versions of the tube were being investigated for Pandora and TV20, they could only use two colours, and Wimbledon looked strange on Red grass - although surprisingly it wasn't noticeable at first - a trick of the mind, but not convincing enough.
I suppose the part of SRL responsible for the games side of the business (our revenue stream) was thinking about what we should do to keep ahead in the market as sales plunged. A small band of individuals within SRL had grouped together and were evolving an idea on paper to rock the market - Loki. A group of SRL employees which excluded Clive - he must have indicated disapproval earlier on. They discovered that I was fairly approachable and this is how I found out about the 'secret' project. I listened and it seemed like a good idea - after all, we had become by chance a games hardware provider but Clive had never felt comfortable with this, at least not in the early days. The QL was an attempt at the professional market, but could our company deliver what was needed. Clive resisted the Loki as I believe he was annoyed that his computer inventions were mostly adopted by the games market, and Loki was a logical way forward if you're in the games hardware market. I think people forget Clive is an inventor, but not a regular inventor or entrepreneur - he's those and more, a modern day Brunnel perhaps would be a better comparison. Think of the everyday products that did not exist before Clive came along....I feel a book coming on.
I produced some sketches. The product simply had a big central processing 'box' into which anything could be plugged into until it fell off the end of the desk or toppled over if you built it skywards. This allowed a low (Sinclair style) entry cost, and could be built up to something big and powerful that would not have been a low entry cost - flexibility. Including flexible and powerful architecture it would have kept us going.
Perhaps Loki could generate revenue to fund Metalab, directed by Clive, to invent new products - that's what Clive likes doing, and he's still doing it today. Figure 18 and 19.

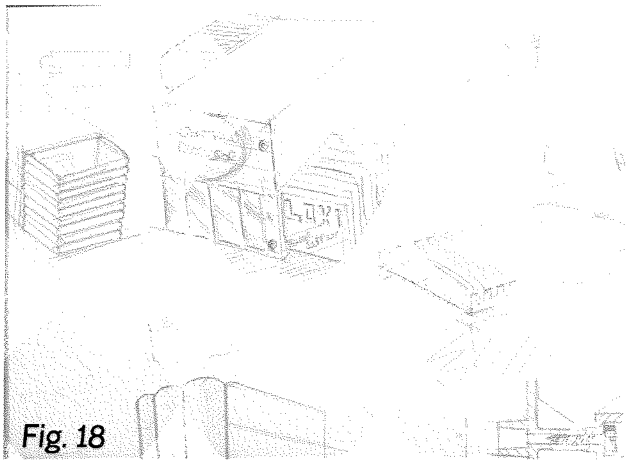


Fig. 18

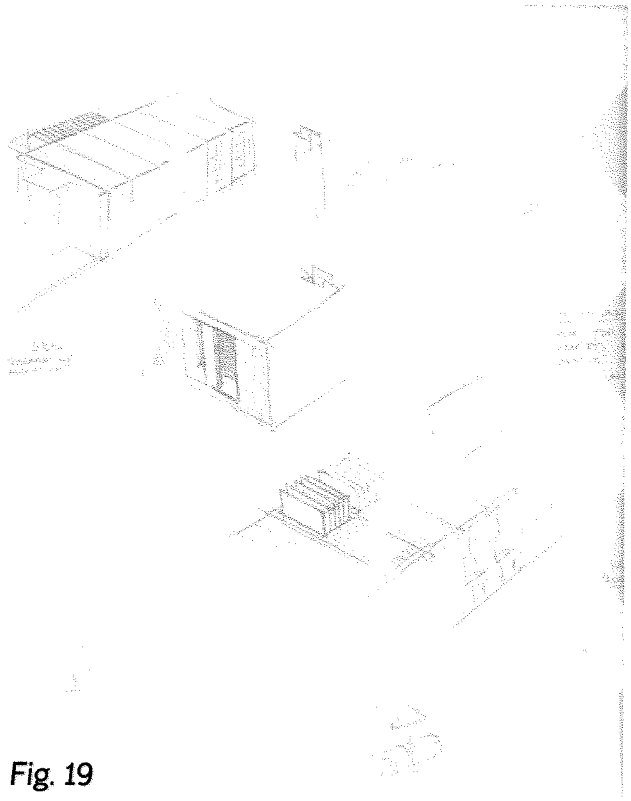


Fig. 19

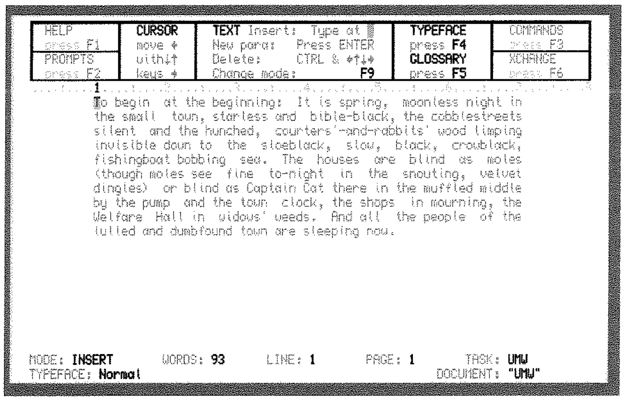
Editor's note: we tried hard to get the pictures in the best print quality possible. Still, some look much better in colour - visit www.flickr.com and enter "Rick Dickinson" - there's more to see!

Tidying Up

by Geoff Wicks

There is a little bit of QL Today that scarcely anyone reads. You will find it about halfway down the right hand column on page 2. Fortunately most of our regular contributors have read it and follow it to the letter, but occasionally someone sends us copy in non-preferred format. We have become quite skilled at converting various text and graphics formats into a form suitable for QL Today. The two that usually defeat us are advertisements sent as PDF documents and Microsoft Word documents containing graphics or unusual formatting.

Sometimes a contributor follows our advice carefully, but we still have a problem reformatting his text. This is best illustrated by an example. Figure 1 shows a simple Quill document. It is the opening paragraph of "Under Milk Wood" by Dylan Thomas typed with left, indent and right margins of 10 characters.



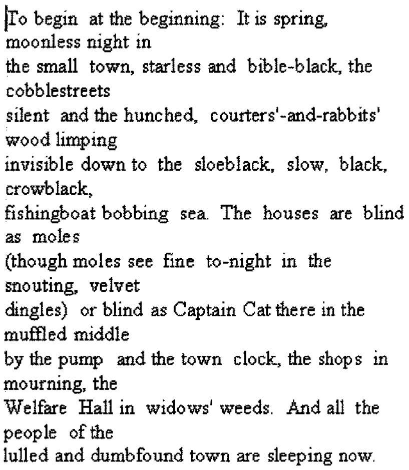
```
HELP        CURSOR    TEXT Insert:  Type at       TYPEFACE    COMMANDS
press F1    move ←    New para:   Press ENTER     press F4    press F3
PROMPTS     with↓↑    Delete:     CTRL & ←↑↓→     GLOSSARY    XCHANGE
press F2    keys →    Change mode:         F9     press F5    press F6

          To begin  at the beginning:  It is spring,  moonless night in
          the small  town, starless and  bible-black, the cobblestreets
          silent  and the hunched,  courters'-and-rabbits' wood limping
          invisible down to  the  sloeblack,  slow,  black,  crowblack,
          fishingboat bobbing  sea.  The  houses  are  blind  as  moles
          (though moles see  fine  to-night  in  the  snouting,  velvet
          dingles)  or blind as Captain Cat there in the muffled middle
          by the pump  and the town  clock, the shops  in mourning, the
          Welfare  Hall in  widows' weeds.  And all  the people  of the
          lulled and dumbfound town are sleeping now.

MODE: INSERT      WORDS: 93      LINE: 1      PAGE: 1      TASK: UMW
TYPEFACE: Normal                                     DOCUMENT: "UMW"
```

Figure 1

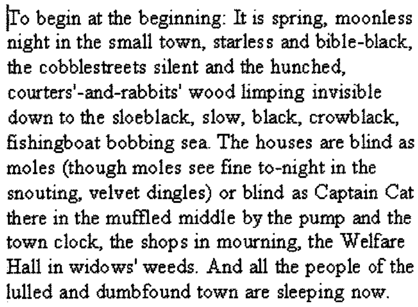
The Xchange version of Quill contains an export routine and our writer uses this to convert the document to ASCII text. When we load this ASCII text into a PC word processor and reformat to the QL Today column width, the result (Figure 2) is not pretty.



To begin at the beginning: It is spring,
moonless night in
the small town, starless and bible-black, the
cobblestreets
silent and the hunched, courters'-and-rabbits'
wood limping
invisible down to the sloeblack, slow, black,
crowblack,
fishingboat bobbing sea. The houses are blind
as moles
(though moles see fine to-night in the
snouting, velvet
dingles) or blind as Captain Cat there in the
muffled middle
by the pump and the town clock, the shops in
mourning, the
Welfare Hall in widows' weeds. And all the
people of the
lulled and dumbfound town are sleeping now.

Fig. 2

Instead our writer could have used my program QL-2-PC Transfer to generate ASCII text. In this case reformatting in a PC word processor is no problem (Figure 3).



To begin at the beginning: It is spring, moonless night in the small town, starless and bible-black, the cobblestreets silent and the hunched, courters'-and-rabbits' wood limping invisible down to the sloeblack, slow, black, crowblack, fishingboat bobbing sea. The houses are blind as moles (though moles see fine to-night in the snouting, velvet dingles) or blind as Captain Cat there in the muffled middle by the pump and the town clock, the shops in mourning, the Welfare Hall in widows' weeds. And all the people of the lulled and dumbfound town are sleeping now.

Fig. 3

If our writer had used Perfection instead of Quill, the export routine built into that program would have given similar problems to Quill, although there is a trick to avoid these. The only QL word processor that would give a similar result to QL-2-PC Transfer ASCII text is Text87.

The purpose of this exercise was not to claim I can write better code than the authors of Quill. That would be both stupid and untrue. The purpose was to demonstrate that ASCII text is not as simple as we sometimes think. It is not just printable characters, but has to contain some control codes. The number, nature and placing of these control codes determines the purpose of the ASCII code.

The difference between the Quill ASCII code and mine is that the former contains 10 line feeds and the latter just one. I suspect the Quill team wrote ASCII code that could be used as a simple printer spooler. I deliberately tailored my code to optimise importation into a PC word processor.

There are several decisions a software author has to make when writing an ASCII text routine.

LINE FEEDS

It is important to understand the distinction between soft line feeds and hard line feeds. In the Quill document there is a line feed at the end of each line. If the margins are changed, then the number and placing of these line feeds will also change. These are soft line feeds. One line feed will not change if the document is reformatted. This is the one at the very end of the paragraph. This is a hard line feed. The Quill ASCII code contains both soft and hard line feeds. Mine contains only the hard line feed.

HYPHENS

Like line feeds hyphens come in soft and hard varieties. In our sample Quill text there are several hard hyphens, but no soft ones. The words "bible-black" are joined by a hard hyphen. The writer intended the hyphen to be there and it would not be removed by reformatting. In my printed copy of "Under Milk Wood" there are several soft hyphens. For example, in the second paragraph, not reproduced here, there is the word "school-teacher". "School-" appears at the end of a line and "teacher" at the beginning of the next line. This is a hyphen that would disappear if the text were reformatted.

Hyphens are not a great difficulty in QL to PC transfers. Quill is the only QL word processor to support soft hyphens and it is easy when scanning a Quill document to distinguish between hard and soft hyphens. Transfers in the other direction are a bigger problem and then a decision has to be made between including or excluding all hyphens.

TABS

These are a major headache for the writer of ASCII text code. In practice there is a choice between 3 options. You can leave tabs out altogether; you can include them and hope for the best; or you can replace them with spaces. Each option has its advantages and disadvantages but, in general, tabs do not work well in ASCII text code. Text87 is versatile when you export text because it offers you the option of either including tabs or converting them to spaces.

Early versions of QL-2-PC transfer ignored tabs completely, but, after a lengthy and persistent argument with François van Emelen at an Eindhoven show, I added them to later versions.

ACCENTED CHARACTERS

This is much less of a problem than in the distant past when printers and telex machines were much simpler and could only handle a limited number of characters. Then the choice was between omitting all accented characters or replacing them with the unaccented version, although at one time German texts replaced the accented character with the unaccented character and then added an "e". Thus a "u" with an umlaut, "ü" became "ue".

Some languages, for example Italian, have characters that are not in the QL character set. In PC to QL transfers QL-2-PC Transfers converts these to the unaccented letter.

In summary if you use a QL word processor and want to send a text file to QL Today, it would help us if you do this using QL-2-PC Transfer. Last year we provided every reader with the full version of this program on a cover disk. If you missed out on this disk you can download the program from the Just Words! web site:

http://members.lycos.co.uk/geoffwicks/justwords.htm

Other Problem Sources

It is not just the occasional writer who gives QL Today a big reformatting job. Most of our news content is extracted from emails and occasionally we use material that has either been scanned by optical character recognition (OCR) or saved as a text file from a website. All these sources can give similar formatting problems. Fortunately the QL has two programs that can make reformatting a lot easier. One of these is QL-2-PC Transfer and the other a Dilwyn Jones program, Tidy-Up.

The QL-2-PC Transfer routines that do this have never been reviewed in the QL press. I have also had no feedback from users other than the people whose asked for their inclusion in the program. The two routines are the OCR text tidy and E-mail message reader. The former is not just for OCR read texts, but can be used for any text that gives reformatting problems.

The OCR text tidy routine attempts to remove soft line feeds, soft hyphens and extraneous spaces from a document. The code that does this is very simple and occupies just 31 lines of basic. The text is scanned character by character and when a hyphen is detected, the following character is examined. If this is a line feed or a carriage return it is assumed that this is a soft hyphen. The hyphen is not printed and the routine scans for the next printable character. Similarly when a line feed is detected the routine checks how many line feeds there are before the next printable character. If there are none then the chances are that it is a soft line feed, and in its place a space is printed. If there are two or more line feeds, these are assumed to be hard line feeds and are retained. Similarly when a space is encountered a check is made to see if the following character is also a space. If so only one space is printed.

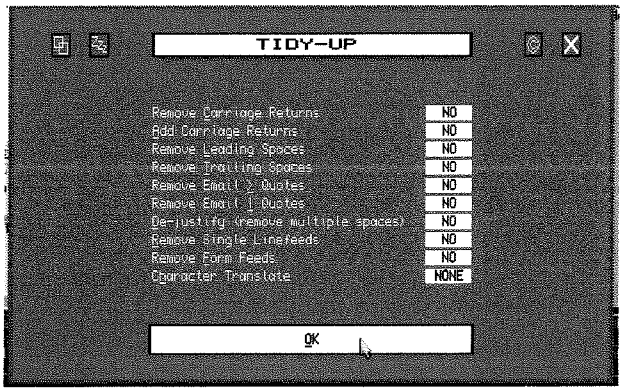
This routine is not totally foolproof. Just occasionally it will wrongly assess a hard hyphen as a soft hyphen. Line feeds are a much bigger problem. The routine relies on there being a blank line between paragraphs and this is rarely so with printed documents. This means that occasionally the routine converts the document into one big paragraph. However, it is still quicker to go through the document reinserting the paragraphs than having to go through it line by line removing the soft line feeds manually.

The other routine in QL-2-PC Transfer attempts to extract the text from an email file and if I were writing the program today I probably would not include it. However I have recently had a good experience of this routine that I shall describe later in this article.

Dilwyn Jones wrote his Tidy-Up program specifically for the needs of QL Today. Dilwyn and I wrote our programs independently of one another although I did provide him with the Windows - QL character conversion codes. It is interesting to see that he had the same ideas as me about how to go about text tidying. Figure 4 shows a screen shot of the program.

Tidy-Up can be downloaded from Dilwyn's website:

http://www.dilwyn.uk6.net/filetran/index.html



```
Remove Carriage Returns                  NO
Add Carriage Returns                     NO
Remove Leading Spaces                    NO
Remove Trailing Spaces                   NO
Remove Email > Quotes                    NO
Remove Email | Quotes                    NO
De-justify (remove multiple spaces)      NO
Remove Single Linefeeds                  NO
Remove Form Feeds                        NO
Character Translate                      NONE
```

Figure 4

Unlike the routines in QL-2-PC Transfer, which are very much take it or leave it, Dilwyn's program gives you more control over the way in which you can modify the text. It also provides you with the statistics of the work it has done. In my opinion its big weakness is that it does not add a space when it removes a soft Line Feed. Like the OCR routine in QL-2-PC Transfer it does transform texts into one large paragraph if there are no empty lines between paragraphs, although on one OCR read text I tried, the places where the paragraphs should be could still be seen. The reason for this was that, unlike QL-2-PC Transfer, Dilwyn's program did not remove Tabs.

E-Mail Problems

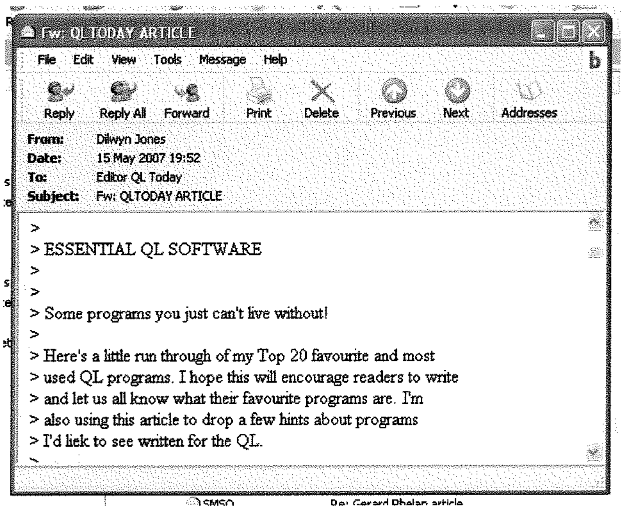
Most of the news items that are published in QL Today have been taken from emails that have appeared on the QL users list. As any subscriber to that list will tell you quite a few of the mailings are messy if a thread has been around for a few days. There is often a lengthy section repeating earlier messages, usually marked by a "›" character, followed by a few sentences giving the writer's contribution to the discussion.

I tested my own and Dilwyn's email tidying routines using 5 emails I have used in QL Today's news section.

Using QL-2-PC Transfer the headers were successfully removed from all emails as were the "›" markers, but some extraneous content at the end of the email remained. It was also necessary to run each file through the OCR tidy routine to get an easily editable text. I should add that this routine was written for emails generated by Outlook Express and I cannot say if it would work for emails generated by other programs.

Dilwyn's program has the advantage that you only have to run an email through the program once to extract the text into an editable form. You also can choose between "›" and "|" for the markers to be removed. This probably means that it is more useful for emails not generated by Outlook Express. However I did find the results less satisfactory than those using QL-2-PC Transfer. Not only did the email header remain, but also the properties section, often more than a page long, that is usually invisible to the user.

Although I wrote above that I was doubtful about the usefulness of the email text extraction routine in QL-2-PC Transfer I have recently had a good experience of its use. One of our regular writers, David Denham, travels around the country in the summer in a camper van. While on his travels he still writes for QL Today, but this year he forgot to take the editor's email address with him and had to send it to us via Dilwyn Jones. As you can see from the illustration (figure 5) the article became seriously corrupted in transport.

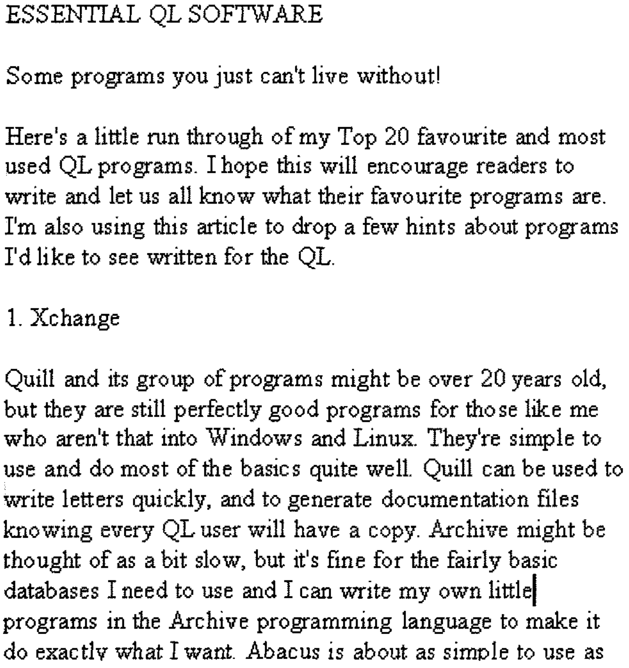


From: Dilwyn Jones
Date: 15 May 2007 19:52
To: Editor QL Today
Subject: Fw: QLTODAY ARTICLE

```
>
> ESSENTIAL QL SOFTWARE
>
>
> Some programs you just can't live without!
>
> Here's a little run through of my Top 20 favourite and most
> used QL programs. I hope this will encourage readers to write
> and let us all know what their favourite programs are. I'm
> also using this article to drop a few hints about programs
> I'd liek to see written for the QL.
```

Figure 5

Reformatting this article manually would have been a long and boring task. It would have meant working through over 400 lines removing the '›' character and soft line feeds. Saving the email as a text file did not help a great deal. However I ran it twice through QL-2-PC Transfer, first using the text extraction routine and then the OCR text tidy routine. In less than two minutes I had a usable text file easily reformatted to the QL Today column width (figure 6).



ESSENTIAL QL SOFTWARE

Some programs you just can't live without!

Here's a little run through of my Top 20 favourite and most used QL programs. I hope this will encourage readers to write and let us all know what their favourite programs are. I'm also using this article to drop a few hints about programs I'd like to see written for the QL.

1. Xchange

Quill and its group of programs might be over 20 years old, but they are still perfectly good programs for those like me who aren't that into Windows and Linux. They're simple to use and do most of the basics quite well. Quill can be used to write letters quickly, and to generate documentation files knowing every QL user will have a copy. Archive might be thought of as a bit slow, but it's fine for the fairly basic databases I need to use and I can write my own little programs in the Archive programming language to make it do exactly what I want. Abacus is about as simple to use as

Figure 6

In summary I have doubts about the usefulness of either program for tidying up emails unless they are extremely long. It is probably just as convenient to save an email as a text file and work from that.

However when it comes to tidying up a text file, whether from a QL word processor, a webpage saved as text or an OCR read text, both programs could be valuable time savers. What is more interesting is that this type of program seems specific to the QL. I know of no PC program that does something similar.

POSTSCRIPT: Jochen has given the editor two light raps on the knuckles as a result of this article.

Firstly he thinks we should make it clear that the article is not intended as a criticism of our contributors. As I wrote in the first paragraph our regular writers follow our guidelines faithfully and cause us few problems. However occasionally an article can become corrupted in transmission or during conversion transfers. Then tidying up programs are useful. They are even more useful when editing the news content.

Secondly he had to correct the article because the editor had used the wrong character for an email marker. Fortunately he kindly spared me the penalty of 100 lines:

"I must not confuse a '‹' with a '›'."

More GPS

by Hugh Rooms

How a GPS receiver knows what time it is

In previous articles I described the Global Positioning System and how it is used to find the accurate location of any place on earth. At the time I wrote them I did not understand the way the relatively cheap clock in the receiver is synchronised with the atomic clocks in the satellites, which is really at the heart of the way it determines its position. I do understand it now.

To summarise, in case you don't remember or have the other articles handy, GPS is a method of location using the calculated distances from a swarm of about two dozen artificial satellites, orbiting at around 12000 miles from the centre of the earth in twelve hour orbits, broadcasting data of time and their positions, on about 1.5 GHz, that allow a suitable receiver to calculate its own position in latitude and longitude, and its height above a datum. They circulate along paths closely monitored and corrected by ground stations, and they have, of course, to be controlled to a greater accuracy than the position you are trying to calculate. The orbits criss-cross in the sky so that any place on earth always has enough in view to get a good fix.

By measuring the times the radio signals take to arrive from the satellites the receiver is able to calculate its distances from them and hence its position relative to them. With a knowledge of the geometry of the orbits this position can be converted into the latitude and longitude of the receiver.

Precisely each second every satellite starts transmitting its sequence of signals. Among them is its own unique identifying signal; this is the PRN or Pseudo Random Number, sent as a long binary number, that I also did not understand then. A good receiver will have a number of processing channels, and, after the initial start up period to find out the time, and what satellites are in view etc., each channel slides (in time) a replica of one of the codes until a correlation is found. So, by this the receiver identifies the satellite, and the slide time gives the delay in receiving its signal, that is the time taken for the signal to travel; divide by the speed of propagation, and you have the distance from the satellite.

The signal, whizzing along at around 300,000 kilometres per second or 30 cm every nanosecond, takes about 68 milliseconds to arrive from a satellite overhead, and from one on the horizon, about 86 milliseconds. Measuring this travel period accurately enough obviously needs a very precise knowledge of the time the radio signal set off, and each satellite has a number of atomic clocks on board, again monitored and controlled from the ground so that they are all synchronised within the satellite and throughout the swarm. However it is difficult to make a correspondingly accurate clock reasonably cheaply so the receiver has to synchronise its own quartz crystal controlled clock to the GPS time, as I will describe in a moment. A pay-off from this is that you can get a very accurate time check – potentially many times more accurate than the MSF transmissions used in radio-controlled clocks, which, with a 60kHz carrier, can give a signal correct only to about one millisecond.

To get a fix, four satellites are needed: the distance from the first defines a spherical 'surface' on which the receiver (strictly its antenna) must be, somewhere. The same applies to a second satellite, so the antenna is now known to be located on the circle where this 'sphere' cuts the first, since it must be on both. A third sphere cuts this circle in two places, and, assuming that the times of travel of the signals, and hence the distances, are accurate, a fourth sphere should pass through just one of these points, giving a unique position relative to the satellites' frame of reference.

With the receiver's clock unadjusted, this fourth sphere will miss the point established by the first three; but they can then be made to coincide by calculating a small correction, and adding it to (or subtracting it from) each measurement: this correction is how much the receiver clock is fast or slow on satellite time. By applying it, the ground receiver's clock is synchronised with satellite time. A succession of these corrections over a few seconds will also give the rate of gain or loss in the receiver's clock.

However the measurements and calculations are not precise, so, in subsequent fixes, even with the corrected clock, the 'spheres' will not meet exactly at the same point. More satellites' distances are measured giving a spread of intersections over a small space called a "resection". A calculation from the points defining the resection gives the most probable position (MPP in the jargon) and a better, average, correction for the clock; also the area of the resection is a measure of the additional inaccuracy of the fix, called "Dilution Of Position" or DOP,above that expected from the inherent tolerances of the system itself.

It is a remarkably clever and ingenious system.

Executable Things - Part 1

by George Gwilt

I describe here how and why I came to use executable Things.

Why

Although setting PROGD$ to the directory containing executable programs can make it easier to load them by omitting the directory when typing the EX command it sometimes fails. In my case this is because I have set PROGD$ to WIN1_C68_ instead of the usual WIN1_SYS_ and then tried to access NET_PEEK from WIN1_SYS_ by typing EX NET_PEEK only to see the message "not found". When this happened several times I recalled that I accessed QD by typing EXEP and that this worked whatever the value of PROGD$. QD is a Thing and is accessed by the keyword EXEP.

I then determined to do the same with NET_PEEK. However, my program would have to be available to those who did not have Things. This meant that I had to arrange for NET_PEEK to be capable of being either loaded by EX or being set up as an executable Thing by LRESPR.

How

First of all we have to know enough about Things for our purpose.

Things are kept in a linked list. Each Thing has an associated linkage block as follows:

Item	Position	Meaning
TH_NXTH	$00	-> next linkage block
TH_USAGE	$04	USAGE list
TH_FRFRE	$08	code called when force remove frees a thing
TH_FRZAP	$0C	code called when thing owner is removed
TH_THING	$10	-> Thing itself
TH_USE	$14	code to USE the Thing, or 0
TH_FREE	$18	code to FREE the Thing, or 0
TH_REMOV	$1C	code to force FREE the Thing, or 0
TH_NSHAR	$20	byte set if Thing not shareable
TH_VERID	$26	version ID
TH_NAME	$2A	name of Thing

All these are long words except TH_NSHAR which is a byte and TH_NAME which is a string.
For all the executable programs I have made into Things I have set the whole linkage block to zero apart from TH_THING, TH_VERID and TH_NAME. The item TH_NXTH must not be zero of course, but it is set by the appropriate linking code as we will see later.
We must now have a look at the Thing itself. I mean by that the Thing to which TH_THING points.

All Things have a header which starts with:

THH_FLAG	$00	"THG%"
THH_TYPE	$04	type of Thing

The type can be -1 to 4 with various meanings. Ours is 1, which means "executable code"

Our Thing continues:

THH_HDRS	$08	offset to code
THH_HDRL	$0C	size of code
THH_DATA	$10	dataspace
THH_START	$14	offset to start of program or 0

The offsets here are all measured from the address of THH_FLAG.
The information given here in the Thing is enough to set up our program as a Job.

The QDOS software for creating a job, MT_CJOB with Trap #1, asks for the length of code and the length of dataspace and also allows an explicit start address to be given. This start address can be to a single version of the code, thus enabling several versions of the program to be running simultaneously with only one copy of the code.
MT_CJOB uses the information it has been given to set up an area in RAM. This starts with a $68 byte header which is followed by an area equal to the sum of the lengths of code and amount of dataspace requested. The header contains a pointer to the start of the program. This points either to the area immediately following the header, or to the start address given to MT_CJOB if this is not zero.
One item relating to the program is not held within the $68 byte header. This item is the program's name. I imagine the reason for this is that names are of indeterminate length so that the header would have had to contain, for the name, a fixed space which will be either too large, which is wasteful or too small which is restrictive. The compromise solution was to set the name 8 bytes after the end of the header which, of course, puts it inside the code, which immediately follows the header. The name is preceded by a word containing the marker $4AFB.
A normal program will thus start with, say, a short branch followed by a long word which is followed by

$4AFB. Then comes a string which is taken as the name of the program which is what will be shown, for example, if you type JOBS.

The program NET_PEEK was written to be re-entrant. That is, it does not alter its own code. This means that multiple copies of the code are not needed for multiple versions of the program.

We can now determine what the Thing linkage block and the Thing itself should contain. Before detailing these contents I should state how the linkage block and the Thing will be set up. This will be done by adding a piece of code to the start of the program. We will CALL this code. In other words we will LRESPR the program to create the Thing. As a consequence the whole of the program's code will have been loaded into a space allocated from the heap.

There is just one more element to be introduced before commenting on the code itself. It is this. Although I determined that NET_PEEK should become a Thing, as I indicated above, I realised also that some machines may not have the Thing code and that therefore NET_PEEK should remain capable of being started by EX. This means that the initial code for NET_PEEK has to decide whether it has been invoked by EX or by LRESPR. Furthermore, it is necessary for safety that a Thing should be set up from master BASIC and not a daughter basic in SMSQ/E. This is because a daughter basic can be removed but the master BASIC cannot.

Comments on the Code below

Initial Code

Whether the program is started by LRESPR or EX the first instruction obeyed is that at HEADZ. This is a normal start of a program with a branch round the name. In this case we go to START where we have to decide how we were born.

We use MT_INF to put our Job ID in D1.L. This will be zero if we were LRESPRd from master BASIC. If so we jump to SET_THING. Otherwise a check is made to ensure that we were not LRESPRd from a daughter BASIC. Each BASIC in SMSQ/E has "SBAS" in -4(A6). Any other non zero ID is taken as arising from EX, in which case we branch to the real start of the program at STARTA.

SET_THING

We set A1 pointing to the linkage block and A0 to the Thing itself.

The software linking in the Thing requires the items from TH_THING to TH_NAME to be filled in, so that is what we do. Most of them are zero for we don't need code to use, free, force free or remove the item.

The Thing itself is filled in with, effectively, the instructions on how the executable program is to be set up when the Thing is called. Thus the code to be set is determined by THH_HDRS, which is set to HEADZ. THH_HDRL is set to PRS - HEADZ, which is the length of code from the start of the program to the end of its name. This is all the "code" we need. The dataspace, $3300 bytes, is set in THH_DATA. Finally, the real start address, STARTA, is set in THH_START.

If the program were not re-entrant the "code" would have to be the entire program, so the length would have to be set accordingly in THH_HDRL. Also the value of THH_START would have to be zero. If this had been done with NET_PEEK, the start would have been at HEADZ and not STARTA. Would this have worked? Yes it would, since the initial code would have decided that the program had not been LRESPRd and a branch would have been made to STARTA as needed.

Linking the Thing

In the operating system SMS2 there are various Trap #1 routines, D0 = $26 to $2C inclusive, which all relate to the Thing system. Alas none of them are available elsewhere, including SMSQ/E. To enable these to be used when these Trap #1s are not available there is, at the end of the list of Things a Thing called THING.

THING contains entries to two vector routines, THH_ENTRY and THH_EXEC. We are interested in the first, since, by jumping to it as a subroutine with the registers set up as for the missing Trap #1s these are implemented.

Code to access the vector is given at GU_THVEC. We use this with D0 = 8. To link in our Thing we need its address in A1, which it is, and D0 set to $26 which we do.

When this is done we return to BASIC.

Code

```
HEADZ    BRA.S    START
         DC.L     0
         DC.W     $4AFB
         DC.W     N_END-NAME
NAME     DC.B     "NET_PEEK V",VERSION," July 2007"
N_END    DS.B     0
PRS

START    MOVEQ    #MT_INF,D0                  ; Set the . .
         TRAP     #1                          ; . . JOB ID . .
         TST.L    D1                          ; . . in D1.L
         BEQ      SET_THING                   ; Master BASIC
         CMPI.L   #"SBAS",-4(A6)              ; Daughter BASIC? . .
         BNE      STARTA                      ; . . No so it was EX
         MOVEQ    #-19,D0                     ; We cannot set the . .
         RTS                                  ; . . Thing from here

SET_THING
         LEA      TLINK,A1                    ; address of linkage block
         LEA      N_THING,A0                  ; address of Thing itself
         MOVE.L   A0,$10(A1)                  ; set address of Thing

; We must link in this Thing

         MOVEQ    #8,D0                       ; THH_ENTR
         BSR      GU_THVEC                    ; Get Thing vector to A4
         BNE      OOPS_1    ---->
         MOVEQ    #$26,D0                     ; Link in . .
         JSR      (A4)                        ; . . the Thing
         TST.L    D0
OOPS_1   RTS                                  ; Back to BASIC

*

; This is the NET_PEEK Thing

N_THING  DC.L     "THG%",1                    ; Marker and type 1
         DC.L     HEADZ-N_THING               ; Offset to "code"
         DC.L     PRS-HEADZ                   ; Size of "code"
         DC.L     $3300                       ; Dataspace
         DC.L     STARTA-N_THING              ; Offset to start of program

; This is the linkage block

TLINK    DCB.W    19,0                        ; Zeroes up to Version
         DC.L     "1.00"                      ; Version
         HED1     <"NET_PEEK">,TLINK1         ; Thing name

; NB HED1 is a macro setting the string < . . > with label TLINK1

; Here is the actual start of the program

STARTA   LEA      (A6,A5.L),A7       set STACK
         LEA      (A6,A4.L),A6       -> DATA SPACE
```

```
   . . .  Program continues  . . .

; Routine to get Thing Vector to A4
; At entry D0.W = $08 for TH_ENTRY routine
;               = $0C for TH_EXEC  routine

VECR      REG       D1-3/D7/A0                 ; Registers to keep . .
GU_THVEC  MOVEM.L   VECR,-(SP)                 ; . . Keep them
          MOVE.W    D0,D3
          MOVEQ     #MT_INF,D0                 ; Get system information
          TRAP      #1
          MOVE.W    SR,D7                      ; Preserve status register
          TRAP      #0                         ; Supervisor mode
          MOVE.L    SV_THINGL(A0),D1           ; address of thing linkage
          BEQ       NOT_THERE ——›
          MOVEA.L   D1,A0
THVEC_LP  MOVE.L    (A0),D1                    ; Next block . .
          BEQ       FOUND                      ; Last block - found
          MOVEA.L   D1,A0                      ; Reset A0 . .
          BRA       THVEC_LP                   ; . . and try again

NOT_THERE MOVEQ     #-7,D0
          BRA       TH_RT

FOUND     MOVEA.L   $10(A0),A0                 ; Pointer to THING Thing
          CMPI.L    #-1,4(A0)                  ; Is it type -1? . .
          BNE       NOT_THERE ——›              ; . . 'fraid not
          MOVEA.L   (A0,D3.W),A4               ; set the vector to A4
TH_RT     MOVE      D7,SR                      ; return to user mode
          MOVEM.L   (SP)+,VECR                 ; replace registers
          TST.L     D0                         ; set condition codes . .
          RTS                                  ; . . and return
```

Directories

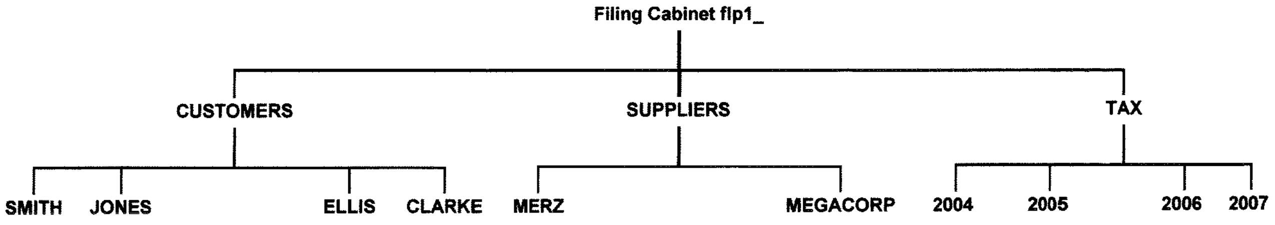
by David Denham

The Toolkit 2 manual says that "a 'directory' is where the system expects to find a file" and goes on to explain that this can be as simple as a drive name like MDV1_ or FLP2_

To help us understand exactly what a directory is, we need to draw a comparison with paper based files.

Suppose our office has a few filing cabinets. All our paper files can be placed in any of these filing cabinets. One is called MDV1_, another is called FLP1_ and another is called WIN1_. These correspond to our microdrives, floppy disk drives and a hard disk respectively. These filing cabinets all have drawers and each drawer contains a number of folders into which the loose bits of paper are grouped. It's obviously convenient to have related files together. For example, one folder per company we deal with.

Grouping papers together like this can make it easier to find something. We have reserved a drawer labelled CUSTOMERS for all our customer correspondence, and another drawer labelled SUPPLIERS for all our suppliers. A third drawer is labelled TAX and contains our tax correspondence for each year in its own folder. We could use a diagram like this to represent the filing cabinet:



Notice how the diagram grows like an upside down tree - the further along the tree you go, the more files there are. Starting from a single stem at the top, it grows additional branches, then those grow additional branches and the whole structure expands like a growing tree. You may have hundreds of sheets of paper in the filing cabinet and if you just placed them on your desk in a random order (like my desk?) you'd have a bit of a problem finding them when you needed them. But when neatly arranged it becomes easier. Want the tax records for 2005? Go to filing cabinet FLP1_, look in the drawer called TAX and pick out the folder called 2005.

This is why directories are often referred to as 'directory trees' because when you draw a diagram they usually end up looking like a tree - one initial stem and then branches spread out, and from those branches individual twigs (or sub-branches) spread out further. When you have a large number of files, they become much easier to handle when grouped into a structure you find easier to cope with. It usually means grouping related files together, so you can focus on those and ignore others.

The concept of 'directories' on a QL is very similar to that of a filing cabinet. Just as a filing cabinet can have individual drawers containing several folders, so our QL drives can have directories on them, each of which in turn can contain several sub-directories or sub-folders. It is traditional to use the term 'directory' on a QL rather than a 'folder', but both mean the same thing. We don't mind which of the two terms you use.

If we were to translate the filing cabinet example above into QL filenames, we'd end up with something like this:

```
FLP1_CUSTOMERS_
FLP1_CUSTOMERS_SMITH_
FLP1_CUSTOMERS_JONES_
FLP1_CUSTOMERS_ELLIS_
FLP1_SUPPLIERS_
FLP1_SUPPLIERS_CLARK_
FLP1_SUPPLIERS_MERZ_
FLP1_SUPPLIERS_MEGACORP_
FLP1_TAX_
FLP1_TAX_2004_
FLP1_TAX_2005_
FLP1_TAX_2006_
FLP1_TAX_2007_
```

And obviously individual filenames would be added to the various directories. Where this comes in useful is being able to list only the files required. So, if we want to list only letters to a customer called SMITH, we could use a DIR or WDIR command to list them:

```
DIR FLP1_CUSTOMERS_SMITH_
```

or

```
WDIR FLP1_CUSTOMERS_SMITH_
```

Note how each directory part ends with a '_' character. This is the official directory separator on a QL, so we can see that FLP1 contains a directory called CUSTOMERS and a sub-directory called SMITH. Unfortunately, the character '_' can also be used in a filename, so it is not just a directory separator. A directory name should end with a '_' (some commands add it automatically).

Default Directories

Suppose we have a large collection of basic programs. It's convenient to have them all in a folder called FLP1_BASIC_. By using the DATA_USE and PROG_USE commands we can tell the QL to use a certain directory by default:

```
DATA_USE flp1_
```

This uses flp1_ as the default drive

```
DATA_USE flp1_basic_
```

This uses flp1_basic_ as the default

So, using the second example, a DIR command with no parameter would list files starting with win1_basic_, so we may have a listing which looks like:

```
mydisk2 450/720 sectors
basic_typewriter_bas
basic_convert_bas
basic_yahtzee_bas
basic_trainer_bas
basic_bills_bas
```

Notice how all my basic programs have the ending "_bas" - I normally add filename extensions to make it even easier to spot file types, for example "_bas" files are basic programs, "_doc" files are Quill documents, "_scr" files are screen pictures and so on. The use of filename extensions is usually (not always) optional as it makes life easier. QL Today has covered the subject of filename endings in the past so I won't dwell on these.

If we wanted to save a new basic program called BACKUP_bas after setting DATA_USE to FLP1_BASIC_, all we need do is SAVE backup_bas, the QL tries to save it as you specified, realises you didn't specify the drive/directory and adds the DATA_USE default setting to it automatically for you, so that it saves it as FLP1_BASIC_backup_bas.

DATA_USE - this default applies to basic programs and general data files.
PROG_USE - this default applies to executable programs.
DEST_USE - this one is a little different, and applies to destinations in commands like WCOPY which have a TO option
SPL_USE - default device for spooling with SPL commands.

I normally keep copies of my executable programs together in one directory called EXECS. This allows me to set PROG_USE to FLP1_EXECS_ and so all I have to do is type EXEC NAME and off it goes, without having to type in the full name or "path" (the path is simply the drive and directory and filename of the file together).
So assuming my FLP1_EXECS_ directory contains:

```
EXECS_QUILL
EXECS_PAINTER
EXECS_TYPER_TASK
EXECS_SIDEWAYS_EXE
```

all I have to do to start Quill is:

```
EXEC QUILL
```

and the QL will realise that there is no drive called QUILL so the EXEC command adds the program default device (as the PROG_USE setting is called) and converts it automatically to

```
EXEC FLP1_EXECS_QUILL
```

The destination default is a little different. It's meant to be used in commands which are sending files from one place to another, such as a copy command. It's meant to provide a default drive and/or directory to be used when nothing is specified as the TO filename. In practice, it's not always used by some QL systems (I don't know why), but there is one more specific use to this destination default device. We have already touched on the notion that a directory name ends with a '_' character. If the DEST_USE command gives a name which ends without a '_' it's treated as a non-directory device, one which does not store files as such, for example, a printer channel. So we could use DEST_USE PAR to ensure that the default destination for copied files is our printer connected to the PAR port:

```
DEST_USE 'PAR'
COPY FLP1_BOOT
```

should then cause the COPY command to send a copy of our boot program to the printer connected to PAR. Note that the names used in the DATA_USE, PROG_USE, DEST_USE and SPL_USE commands can be quoted or unquoted, and are not case sensitive (FLP1_BASIC_ is the same as flp1_Basic_ for example).

Checking the Defaults

There are 3 functions to check the current settings of the three defaults.
PRINT DESTD$ prints the string DATA_USE is set to.
PRINT PROGD$ prints the string PROG_USE is set to.
PRINT DESTD$ prints the string DEST_USE is set to.
Here is a short superbasic program to illustrate how to check and set these defaults. This is quite useful in itself - you could compile this with Turbo and set it up on a hotkey to be there when needed.

```
100 REMark set DATA_USE,PROG_USE and
    DEST_USE
110 CLS : CLS #0
120 PRINT'DATA_USE=';DATAD$
130 PRINT'PROG_USE=';PROGD$
140 PRINT'DEST_USE=';DESTD$
150 INPUT\'New value for DATA_USE >
    ';datause$
160 INPUT 'New value for PROG_USE >
    ';proguse$
170 INPUT 'New value for DEST_USE >
    ';destuse$
180 DATA_USE datause$
190 PROG_USE proguse$
200 DEST_USE destuse$
```

There's also a command called DLIST in Toolkit 2 which lists the default settings for all three:

```
DLIST #channel_number
```

So DLIST #1 will send the short list of the three default values to screen window channel #1.

LEVEL 1 and LEVEL 2

You may have heard of the term "level 2 directories" which is used with newer disk systems and hard disk systems.
The original QL disk systems used something called a "level 1" directory structure (also called 'soft' directories). Basically, these weren't really directories, they simply treated filename prefixes as though they were directories and although still useful, they didn't really handle directories in any way in which users of other computers would recognise.
The main difference was that when you called up a list of files, you saw all files rather than a list of directories. The filenames were not 'contained' in the directory as such, the filenames were just

identified by name prefixes and did not really correspond exactly to our comparison with an office filing cabinet.

So, using our example above, when we did a DIR FLP1_ we got a complete list of files like this:

```
mydisk2 450/720 sectors
CUSTOMERS_
CUSTOMERS_SMITH_
CUSTOMERS_JONES_
CUSTOMERS_ELLIS_
SUPPLIERS_
SUPPLIERS_CLARK_
SUPPLIERS_MERZ_
SUPPLIERS_MEGACORP_
TAX_
TAX_2004_
TAX_2005_
TAX_2006_
TAX_2007_
```

But the intention was to create three independent directories on the disk, where the files would be separated into independent groups, so ideally our DIR command should have given us a result as follows:

```
mydisk2 450/720 sectors
CUSTOMERS
SUPPLIERS
TAX
```

In other words, if we do a DIR of the filing cabinet itself, we only want a list of the drawers, not of every bit of paper in all of them.

If we then wanted a list of files in a given drawer, we could do something like DIR FLP1_CUSTOMERS_ which would give us a list of the folders or sub-directories contained within CUSTOMERS (rather like a list of drawer content attached to the front of each drawer of our filing cabinet):

```
mydisk2 450/720 sectors
CUSTOMERS_SMITH_
CUSTOMERS_JONES_
CUSTOMERS_ELLIS_
```

Having got as far as this, we can then list the individual files within each of these customers' sub-directories. Hopefully, it will be clear by now that we can go up and down the "tree" as we wish and it should also be clear how storing groups of files together makes it easier to locate them by keeping such groups down to more manageable sizes, especially when we have such large storage capacities available to us these days and QL software is usually fairly small compared to, say, Windows packages.

Files which are not placed in any sub-directory of the disk are referred to as being in the "root directory", i.e. saved direct onto the drive itself with a command such as SAVE FLP1_BOOT.

This is the difference between Level 1 and Level 2 directories. Basically, level 2 directories are separate locations for filenames. You can put groups of files into a directory and they will only be listed when you are handling that particular directory.

Here's a short list of examples of which systems use Level 1 and Level 2 directory systems. This list is not exhaustive, these are only examples of systems I think I know about. In general, if your system has a MAKE_DIR command to create real directories, it is level 2.

LEVEL 1

- Unexpanded QL
- interface systems, e.g. QDISK
- Most pre-Trump Card disk
- Miracle Systems Trump Card
- Older ramdisks
- Some emulators, e.g. QLAY
- QemuLator for Apple Macs
- QDOS Classic Amiga

LEVEL 2

- Qubbesoft Trump Card
- Gold Card
- Super Gold Card
- QXL
- Modern ramdisks
- QemuLator for Windows
- Miracle Hard Disk
- QPC1 and QPC2
- Qubide
- Q40 and Q60

Creating Hard Directories

There's a command and a function on most Level 2 systems to create the directories on the first place:

```
MAKE_DIR FLP1_BASIC_
```

or

```
LET any_error = FMAKE_DIR(FLP1_BASIC_)
```

The function version makes it easier to apply error trapping, for example, allowing a second try when someone tries to create a directory but forgets to insert a floppy disk:

```
100 REPeat loop
110 INPUT'Enter name of directory > ';d$
120 any_error = FMAKE_DIR(d$)
130 IF any_error < 0 THEN
140 REPORT #1,any_error
150 ELSE
160 EXIT loop
```

```
170 END IF
180 END REPeat loop
```

That example will not work on version AH or JM QL ROM because of the use of the REPORT command.

You can create the directories which represent the drawers in our filing cabinet with a few MAKE_DIR commands:

```
MAKE_DIR FLP1_CUSTOMERS_
MAKE_DIR FLP1_SUPPLIERS_
MAKE_DIR FLP1_TAX_
```

The sub-directories they contain can be created in two ways, explicitly using the full path names after you have created the initial directory, or less directly by setting the DATA_USE default to the directory required after creating the initial directory then issuing a series of MAKE_DIR commands with the new sub-directory names:

1. Explicitly

```
MAKE_DIR FLP1_CUSTOMERS_
MAKE_DIR FLP1_CUSTOMERS_SMITH_
MAKE_DIR FLP1_CUSTOEMRS_JONES_
MAKE_DIR FLP1_CUSTOMERS_ELLIS_
```

2. Indirectly

```
MAKE_DIR FLP1_CUSTOMERS_
DATA_USE FLP1_CUSTOMERS_
MAKE_DIR SMITH_
MAKE_DIR JONES_
MAKE_DIR ELLIS_
```

Both methods give the same result, but one method may be more useful than the other in some contexts.

Another difference with level 2 systems is that most systems list the directory names with a marker alongside the name, usually a ' ->' after the directory name, e.g. when the three directories created above exist alongside a program called flp1_BOOT:

```
DIR FLP1_

450/720 sectors mydisk2
CUSTOMERS ->
SUPPLIERS ->
TAX ->
boot
```

So this example tells us that there is a file called 'boot' in the "root" directory, and three directories. Note that directory names are often listed without the '_' ending. In fact, many directory-related commands such as DATA_USE or MAKE_DIR will usually append the '_' to a directory name automatically as and when required.

Note the restriction on directory and filename lengths on QL system. In general, they are restricted to 36 characters, including both directory and filename, so you couldn't have, for example, a 36 character filename contained within a 36 character directory name as that would make the total length 72 characters. The 36 character limit doesn't include the drive name, so that could add another 5 characters making the total combined length limit of 41. This is not too great a restriction in most cases, since very long path names quickly become difficult to handle. The only time it's been a problem in my experience is when trying to unzip archives create using ZIP on other computers. As an example, try unzipping Jonathan Hudson's Lynx package into a subdirectory on a QL system - you quickly run out of characters unless it's unzipped to the root of a hard disk, which is inconvenient.

Directory Navigation

Toolkit 2 includes a useful set of 3 commands for navigating along directory trees. These are **DDOWN**, **DUP** and **DNEXT**, which work by adjusting the DATA_USE defaults. To visualise how they work, we need to think of the example tree listed above to see how they move along the structure. Suppose we start off with DATA_USE FLP1_, then issue a DDOWN "suppliers" command. The DATA_USE value is now FLP1_suppliers_. Next, we issue a DDOWN Merz command. The DATA_USE value now becomes FLP1_suppliers_Merz. But next we wish to go back up the tree, back to suppliers, so to go back to the next level up the tree we issue a DUP command to Directory UP one level, in other words DATA_USE is now FLP1_suppliers. Issue another DUP command and it goes back to FLP1_.

DNEXT moves to another directory at the same level as the one we are in at the moment. So if we are in FLP1_suppliers_ and wish to jump to FLP1_tax_ we can simply issue a DNEXT 'tax' command to do a sideways jump within the same level.

Commands

With Toolkit 2, the following file maintenance commands use the DATA_USE default to various extents (many are Toolkit 2 commands):

DIR, WDIR, STAT, WSTAT, DELETE, WDEL, COPY, COPY_O, COPY_N, COPY_H, WCOPY, SPL, SPLF, RENAME, WREN

The various file saving, loading and opening commands also use the DATA_USE default as long as the system has Toolkit 2

LOAD, SAVE, MERGE, MRUN, OPEN, OPEN_IN, OPEN_NEW, SBYTES, LBYTES, LRESPR, OPEN_OVER and **OPEN_DIR**
Commands such as **EX, EW, EXEC** and **EXEC_W** which execute programs use the PROG_USE default instead.

Listing the Directories

If you have the WD extensions from Phil Borman (they were supplied on the Qubide utilities disk as a file called WD_BIN and you can download them from some QL websites too), these include a handy little command called TREE which lists the directories on a hard disk, starting from the current DATA_USE setting. The output looks like this:

```
CUSTOMERS
CUSTOMERS_SMITH
CUSTOMERS_JONES
CUSTOMERS_ELLIS
SUPPLIERS
SUPPLIERS_CLARK
SUPPLIERS_MERZ
SUPPLIERS_MEGACORP
TAX
TAX_2004
TAX_2005
TAX_2006
TAX_2007
```

If, like me, you'd prefer a listing whereby only the sub-directory names are shown in the listing, with the sub-directory names indented a little, here is a little superbasic program to do this, based on a listing in QL Today Volume 3 Issue 5. You can tinker with this listing until the cows come home - some hints are included to enable you to customise it as you require.

It uses a version of Dilwyn Jones's Extended Dir routine to recursively find directory names - once in a given directory, scan through it for further sub-directory names and call it again to scan those. So, using our filing cabinet analogy, we start with a given filing cabinet and check how many drawers it has. Starting with the first drawer, we look into that to see how many folders it contains. Once we've finished that drawer, we move on to the next - any empty drawers only result in no sub-directory names being printed.

The program produces an output listing to the screen, pausing every 20 lines (press any key to continue). If you wish to modify it to print the tree instead, remove lines 1070 and 1250, and change the PRINT commands to output to a specified channel number instead, e.g. 1075 OPEN #3,'SER1' and change the PRINT commands in lines 1230 and 1240 to PRINT #3, instead.

The amount of indent for each sub-directory name is specified in line 1260, where the number added to the variable "indent" is the depth of indent for each sub-directory branch.

The directory names printed don't end with '_', since the QL doesn't seem to store them with the '_' - I presume that this is because '_' is 'assumed' as separator by convention. If you prefer to see the '_' character after each directory name, just add it at the end of the PRINT statement in line 1240.

Some QL file handling programs like the QPAC2 files menu precede directory names with a symbol such as a '›' character. This is quite easy to implement by changing the PRINT command in line 1230 to PRINT FILL$(' ',indent);'› ';

```
1000 REMark Directory Tree Listing by David Denham
1010 REMark based on Dilwyn Jones routine in QL Today Volume 3 Issue 5
1020 :
1030 CLS : CLS #0
1040 INPUT #0,'Drive › ';dr$
1050 INPUT #0,'Directory › ';drc$
1060 PRINT dr$
1070 line_no = 0
1080 Show_Tree dr$,drc$,0
1090 :
1100 DEFine PROCedure Show_Tree (drive$,directory$,indent)
1110 LOCal loop,channel,name$,dir_position
1120 channel = FOP_DIR (drive$&directory$)
1130 IF channel ‹ 0 THEN RETurn
1140 dir_position = 14
1150 REPeat tree_loop
1160 BGET #channel\dir_position
1170 IF EOF(#channel) THEN CLOSE #channel : EXIT tree_loop
1180 GET #channel,name$
1190 IF LEN(name$) › 0 THEN
```

```
1200 REMark a directory length of 0 may be a deleted file
1210 BGET #channel\dir_position-9 : REMark file type byte
1220 IF CODE(INKEY$(#channel)) = 255 THEN
1230 IF indent › 0 THEN PRINT FILL$(' ',indent);
1240 PRINT name$(LEN(directory$)+1+(indent›0) TO LEN(name$))
1250 line_no = line_no+1 : IF (line_no MOD 20) = 0 THEN PAUSE
1260 Show_Tree drive$,name$,indent+2
1270 END IF
1280 END IF
1290 dir_position = dir_position + 64
1300 END REPeat tree_loop
1310 END DEFine Show_Tree
```

Example of the output from this program:

```
CUSTOMERS
SMITH
JONES
ELLIS
SUPPLIERS
CLARK
MERZ
MEGACORP
TAX
2004
2005
2006
2007
```

The Network

There is a slight difference when accessing files over the QL network when Toolkit 2 is used. Suppose you are accessing files stored in a directory called SUPPLIERS_ on flp1_ on network station 6.

```
DIR n6_FLP1_SUPPLIERS_
```

Here, the drive is n6_ and in many ways the sub-directory is FLP1_SUPPLIERS_. I'm not going to hark on too much about the network here, just mention that it is something you need to be aware of when using the network. Just be aware that sometimes a "directory" over a network can actually be a whole drive. In many ways, the remote network station is a directory as far as your QL is concerned and the various drives can be just sub-directories - but this is getting rather complex.

Conclusions

Directories are a powerful resource on your computer, but not the easiest of subjects to master. You may be able to get away without using them on a simple floppy disk system, but once you start using hard disks, be they Qubides or QXL.WIN emulator systems you will pretty soon wish you knew about directories when you see the number of files build up!

I have made a number of comparisons in this article with filing cabinet and paper based offices, and to me this is the best way of thinking about directories - too many bits of paper cause confusion without a proper filing system. The same is true of the number of files we can have on QL systems nowadays.

Programming in Assembler - Part 19

by Norman Dunbar

Greetings from the basement!

We have moved house and are getting settled. We have still got a lot of boxes to unpack and things to find, but we are getting there. I have a new 'office' deep down in the basement where it is nice and cool. This is the first in the Assembler series to come from the basement.

With all the upheaval of getting moved and unpacked etc, I have not got a lot of code for you this time, hopefully, you won't be too bored by this episode in which I go over bits and pieces of assembly language programming that causes me grief.

It all started when I was having a think the other day about life in general and assembly language in particular. I was pondering on the bits of programming in assembler that I always get wrong, or have to really think about – and still get wrong.

SIGNED and UNSIGNED tests

I don't know about you, but I seem to have severe difficulties in remembering which are the signed and which are the unsigned tests. I have to confess that I always have a list of them written down (or printed out) and stuck to my work area – wherever that happens to be.

Here is a reminder of the 'cc' code to use in a Bcc or whatever for signed and unsigned comparisons:

Test	Signed	Unsigned
Greater Equal	GE	CC
Greater Than	GT	HI
Equal/Zero	EQ	EQ
Not Equal/Zero	NE	NE
Less Equal	LE	LS
Less Than	LT	CS
Negative	MI	—
Positive	PL	—

So, if D0.B contains the value $FF it represents either 255 (unsigned) or -1 (signed). You, as the programmer should know whether the value is considered signed or not and can make the correct comparison checks.

The EQ and NE tests are interesting in that they either mean 'two values are [not] the same' when comparing things such as memory and regiters, or two registers etc, or, when having just loaded a register with a value, they mean 'the value just loaded into a data register is [not] zero'.

The following code examples are identical in result, one is just quicker than the other:

```
MOVE.W (A1),D0
BEQ.S  D0Zero
....
```

and

```
MOVE.W (A1),D0
CMPI.W #0,D0
BEQ.S  D0Zero
...
```

Which way round is the 'subtraction' in a CMP instruction

If I see CMPI.W #1234,D0 then it is obvious, I am comparing D0.W with the value 1234. That's easy. However, when I see CMP.W D0,D1 I lose the plot. What am I comparing here is it D0 with D1 or the other way around. My brain hurts already. Is the value of 1234 subtracted from D0 or is the value in D0 subtracted from 1234. Which way round is the subtraction and the resulting setting of flags?

The answer, I note from part 2 of this series is that the source register is subtracted from the destination register exactly as a SUB instruction would do, the result is simply discarded. So in the instruction CMP.W D0,D1 the flags are set according to D1.W minus D0.W.

It is assumed that after this pseudo-subtraction, some Bcc, Scc or DBcc instruction will no doubt check the flags and do something useful with the result.

Which CC code to use after a CMP

Leading on from the above, I never remember which 'cc' code to use after a CMP – although, having written out the above it is becoming clearer. The following code gives me the willies time after time:

```
...
CMP.L D0,D1
BHI   somewhere
...
```

This fragment has everything that confuses me, almost. It has a CMP followed by a 'cc' instruction – so I have to think about the two 'problem areas' I mention above. Signed or unsigned and which register is causing the HI to be true or false.

Well, the HI is, from my table above, unsigned and using my new found knowlege of the CMP instruction I know (for a short while at least) that the flag are set to the result of (D1.L - D0.L) but which way around does it go again ?

The BHI should be read as "branch if destination register HI source register" in the preceeding CMP or SUB or whatever was used to set the flags. So, using this explanation, I now know that the code above branches if D1.L is higher (in an unsigned manner) than D0.L.

This leads me to surmise that the following pseudo-code:

```
IF unsigned (D1.L > D0.L) THEN
...
ELSE
...
END IF
```

Becomes:

```
IF       CMP.L D0,D1
         Flags = result of D1.L - D0.L
         BHI.s THEN
         D1 is indeed GT (signed) than D0
ELSE     ...
         D0 is less or equal to D1 (the ELSE bit)
         BRA.s ENDIF
         Skip over the THEN clause
THEN     ...
         Do the THEN stuff
ENDIF    ...
         Together again.
```

Alternatively, reverse the jumps to look more like the pseudo code:

```
IF       CMP.L D0,D1
         Flags = result of (D1.L - D0.L)
         BLS.s ELSE
         D1 is not HI (unsigned) than D0
THEN     ...
         D1 is indeed HI than D0, (the THEN bit)
         BRA.s ENDIF
         Skip over the ELSE clause
ELSE     ...
         Do the ELSE stuff
ENDIF    ...
         Together again.
```

Maybe we should think about writing our assembly language in pseudo code and having a pre-processor convert it into the real assembler code ....

What about loops with conditions

The instruction format for decrement and branch on condition instructions is DBcc where 'cc' is one of the many condifion codes noted above.

So, you have an area of RAM full of data and you go looking through it for the first occurrence of a specific byte value, let's say $00, and you know that the leading word of the data defines the length in bytes. So, the following fragment would do the job - assuming A0.L points to the data and D1.W holds a valid data length.

```
LOOP     CMP.B #$00,(A0)+
         DBcc D1,LOOP
ENDLOOP ...
```

What we need to figure out is which 'cc' we require and also, what is result when we get to the ENDLOOP label if we found a zero byte or if we didn't.

One way we will end up at ENDLOOP is when our counter in D1 expires - reaches minus 1 - that indicates that we ran out of data before finding what we wanted. But, what happens if we find a zero byte - and which 'cc' do we need.

If we remember that DBcc really means 'test condition and decrement if false and branch' then we should be ok. Alternatively :

```
IF 'cc' is FALSE THEN
  D1 = D1 - 1
  IF D1 <> -1 Then
    GOTO LOOP
  ELSE
    GOTO ENDLOOP
  END IF
ELSE
  GOTO ENDLOOP
END IF
```

So, we want to check for a zero byte, we can use the 'EQ' test - remember EQ means we have hit a zero or two values are not equal - and our code now becomes :

```
LOOP     CMP.B #$00,(A0)+
         DBEQ D1,LOOP
ENDLOOP ...
```

So, we have reached ENDLOOP and we need to know if we hit a zero byte or if we ran out of data. How to tell?

Well the good news is that the DBcc instructions do not alter the flags. SO on exit from a DBcc loop, if the 'cc' is still true, then the condition was met and the loop terminated before the counter ran out. All we have to do is retest with the same condition as follows

```
LOOP     CMP.B #$00,(A0)+
         DBEQ D1,LOOP
ENDLOOP  BEQ.S FoundZeroByte
```

In this case, we check for the EQ condition which tells us that the loop terminated early.
We can test the inverse condition as well to see if the loop expired without hitting therequired condition:

```
LOOP     CMP.B #$00,(A0)+
         DBEQ D1,LOOP
ENDLOOP  BNE.S NotFound
```

Which we see makes the branch if the loop expired when the counter in D1.W hit minus 1. I propose that we rename this family of instructions to 'Decrement and Branch UNLESS condition'. That makes more sense to me.

Do I have to TST.L D0 after a TRAP or a Vector call

I always get corrected on this one, either by George or Simon. For years I have always done this:

```
...
TRAP #1
TST.L D0
BNE HandleError
...
```

Which is fine for a TRAP call – it has to be done this way. However, for a vector call it is different:

```
...
MOVE.W UT_GTSTR,A2
JSR (A2)
TST.L D0
BNE HandleEror
...
```

This is wrong – I do not need to test D0 after a vectored utility call. The reason I do after a TRAP and don't after a vector is quite subtle and was only recently pointed out to me by Simon when it all became very clear indeed.
A TRAP call is treated as an exception and to return from an exception handler, you use the RTE instruction. To return from a vectored call, it is an RTS instruction. The difference between the two is that the RTE restores the status register as well as the program counter. RTS simply restores the program counter.
So, all these years where I've been testing D0 on return from vectors I've been wasting clock cycles when I need not have done. The status register is correctly set on exit from a vectored utility but has only D0 is set on return from a TRAP.
Simple, but it has caught me out for years. I now need to unlearn my habit of coding a TST.L D0 every time I use a vectored utility.

Happy coding.

SuperBASIC Versions

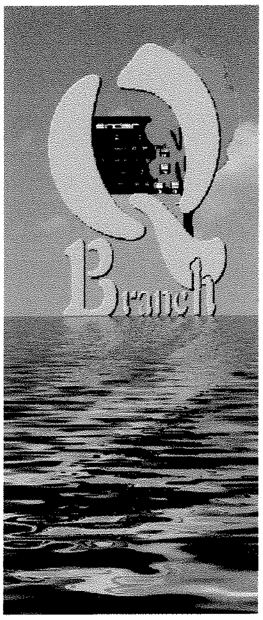
by Dilwyn Jones

SuperBASIC is the BASIC interpreter built into QDOS ROMs, including Minerva, although the Minerva version has a few enhancements. The SMSQ/E operating system comes with an enhanced BASIC interpreter, called SBASIC. Over the years, the term S*BASIC has been adopted to mean either or both versions of QL BASIC.

The SuperBASIC interpreter was originally developed by Jan Jones at Sinclair, while QDOS itself was developed by Tony Tebby. The original plan had been to use an operating system commissioned from a company called GST, but for a variety of reasons this was never supplied with the QL and Sinclair used Tebby's QDOS instead.

The GST operating system, called 68k/OS, was marketed by GST as a plug in card for the QL for a while, though it didn't sell in great numbers.

The Minerva ROMs were originally developed by a small team at QView – Jonathan Oakley, Stuart McKnight and Laurence Reeves, and the initial letter of their forenames gave the SuperBASIC version of "JSL1". Minerva was originally based on QDOS ROMs, although the many changes and improvements made to Minerva over time meant they came to be recognised as separate ROMs in their own right.



Feeling out on a Limb?
Reach out for QBranch
Suppliers of Quality QDOS/SMSQ products
Hardware and Software

20 Locks Hill
Portslade
Sussex
BN41 2LB
UK
Tel: +44(0)1273 386030
Fax: +44(0)1273 430501
Mobile: +44(0)7836 745501

email : sales@qbranch.demon.co.uk
www.qbranch.demon.co.uk

We can accept payment by Visa, Mastercard, Maestro and Switch. We also accept sterling cheques drawn on a UK bank. or payment direct to our bank:
Natwest
Sort code 60-17-01
Acc. 84534826
BIC : NWBKGB 2L
IBAN :
GB34 NWBK 6017 0184 5348 26

VISA MasterCard Maestro

QDT

The Ultimate QL Desktop

v1.02

£41.50 + p&p

£2.00 UK / 2.50 EU / £ 3.00 RoW
Needs SMSQ/E & extended colours to run

EasyPTR v4

Pts 1&2
Major Rewrite!

£42.00

Produce High Colour pointer-driven menus from SuperBASIC

Also on CD Rom

Text 87 £ 79.00

High Colour Patch (all systems) £9.00
ESC/P2 drivers £ 26.00 / Fountext £39.00
Typset94 / 2488 drivers £29.00 ea.

SMSQ/E and QPC 2

SMSQ/E for Gold Card / Atari / QXL / Q40
Aurora High Colour Driver included with Gold Card Version

Now Only **£15.00 + postage**

QPC 2 v 3.33

Now only £ 42.00

Upgrade Prices:
from v 3.xx Free
from v 2.xx £13.90
from v 1.xx £34.00

Delivered on CD ROM with Manual as Adobe Acrobat File

QPCPrint

Print to any Windows Installed Printer from Your QL Programs !

£27.00

Programs upgraded to high colour version

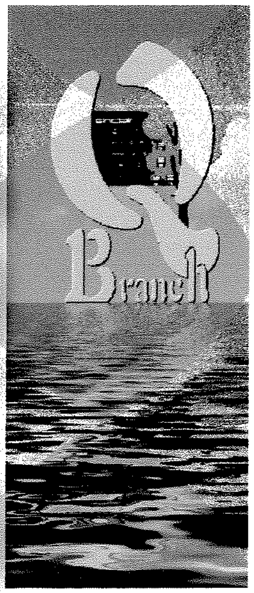
QMake / WinEd / QPAC 1 / QPAC 2 / QSUP / SuQcess / Disk Mate 5
Send old master disk and £1.60 per disk to cover costs

Free !

Paid Upgrades	
QD 98 to QD 2003	£10.50
QSpread 2001 to QSpread 2003	£10.50

Return* Master Disk with *order

Epson EPL-6200 Printer. QL compatible Paralell Port) with 1 new toner.
£75.00 + Carriage
Samsung 19” CRT Monitor
Sony Triniton Flat Screen
£50.00 Collection advised.



Programs

Utilities

Fifi2	File Finder	£21.00
QSup	Utilies	£30.00
QSpread	Spreadsheet	£51.00
Cueshell 2	File Manager	£15.00
QPAC 2	File Manager & Utilities Package	£42.00
QPAC 1	Calendar, Clock, Calculator, Sysmon	£22.00
QLoad/Qref	Fast load for QDOS	£15.00
QTYP2	Spell Checker	£31.00
QLQ	Printer Utility	£30.00
SuQcess	Database	£28.00
Q-Route	Route Finder	£25.00
Knight Safe3	Backup Program	£35.00
Fractal Collection	Fractals	£35.00
QCount	Accounting	£25.00

Programming

QD 2003	Text Editor & More	£ 49.00
QBASIC	QLiberater to QD Link	£ 15.00
QLiberator	Basic Compiler	£ 50.00
QD + QBASIC		£ 63.00
QD + QBASIC + QLiberator		£104.00
QPTR	Pointer Toolkit	£ 32.00
MasterSpy	Fast Text Editor	£ 30.00
QMake	Assembler Tools	£ 18.00
QMon/Jmon	Monitor - Upgrade only	£ 22.00
BASIC Linker	Basic Library Linker	£ 22.00
Disa 3	Dissassembler	£ 34.00
QMenu	Menu Extensions & tutorial	£ 16.00
Easyptr v4	Toolkits & Programming Extns	£ 41.50
Easyptr v4	Part 3 C extensions	£ 14.00

HARDWARE

We have a rotating stock of both new and second user hardware. It is best to call or email us for details of what is available.

Recycled Items
(when available)

Super Gold Card	£110.00
Gold Card	£ 45.00
Aurora	£ 65.00
QXL	£ 35.00
superHermes	£ 65.00
DiRen Keyboard Interface	£ 15.00
Qplane	£ 5.00

New Items

Aurora	£70.00
Aurora w/8301 & Minerva	£80.00
8mb Rom Disq	£98.00
4mb Rom Disq	£65.00
2mb RomDisq	£39.00
mPlane	£34.00
MCplate	£ 6.50
Various braQuets	£ 8.00
Gold /Super Gold Card Batteries	£10,00

We also have a collection of standard QLs, QL Power supplies and some QL books.

Cables for the Aurora, Qubide and Super Gold Card ROMs and other QL accessories are also available from us.

Call for details

The Argos ROMs for the Thor computers were developed by David Oliver at CST, the company behind the Thor computers. Argos is based on QDOS, but includes many system enhancements designed specifically for the Thor systems.

SMSQ/E was developed by Tony Tebby with more recent versions being maintained by Marcel Kilgus and the SMSQ/E Registrar Wolfgang Lenerz, who co-ordinates releases of the various versions for the various platforms. As an interesting aside, there had been an earlier version called SMS2, which was only available as a plug in cartridge for Atari systems. Miracle Systems commissioned an operating system called SMSQ from Tony Tebby for their QXL Card. This was a predecessor of SMSQ/E, the major difference between SMSQ and SMSQ/E is that SMSQ does not include any pointer environment (the /E in SMSQ/E stands for the Extended Environment built into that operating system). As for the name SMSQ itself, various guesses have been made as to what the letters stand for, the likeliest explanation being that it stands for Single-user Multitasking System for QL. SMSQ and SMSQ/E are normally loaded from disk rather than being installed in ROM, although the Q40 can store its version of SMSQ/E in onboard flash ROM if required.

Versions of BASIC are indicated by a 2, 3, or 4 character string such as AH, JM, JS, MG, JSL1 or HBA. Minerva uses "JSL1" for SuperBASIC in all its various ROM versions, while SMSQ/E always uses "HBA" as the SBASIC version. QL ROM versions have generally been known by these versions (e.g. QL ROM version JS) although strictly speaking they refer to SuperBASIC versions. The operating system version number has generally been indicated by a 4 character version number such as 1.10, with the '.' replaced by a letter for national variations (e.g. QDOS version 1E13 with SuperBASIC version MGE for a Spanish QL ROM)

The version of BASIC can be checked with the command

```
PRINT VER$
```

which will print the two, three, or four character version string. Checking the QDOS version is not quite so easy as there is no BASIC command or function to establish that version. Some toolkits do include a function such as QDOS$ which will return the four character identifier string. Alternatively, just view a ROM image on the screen or search through it for a string starting with "1." or "2." – if not an English ROM version, you may need to replace the '.' with the national letter for your country. Those familiar with machine code can use the MT.INF trap call (trap #1, d0=0) to return the QDOS ID string as a long word in the D2 register.

The first release version of the QL ROM was known as version FB – this was somewhat bug-ridden and incomplete, so the FB probably stood for Full of Bugs! This was quickly replaced by version PM which was an improvement, though still not really suitable for a finished version. Version AH was the first version which could be considered stable enough for permanent use, and this was quickly followed by version JM, the first version to be supplied on a ROM as opposed to earlier versions released on EPROM.

Early in 1985 version JS was released and this contained significant improvements, such as error trapping, TRA character translate systems for printing, and variable watching (WHEN var). Version JS fixed some bugs in earlier versions, but also introduced a few obscure new bugs of its own. Version MG was released a little later for overseas markets – national ROM variations, such as MGF for France, MGE for Spain and MGG for Germany. Sinclair did not release a version MG for the UK market, although a privately produced MGUK ROM does exist.

A few private ROM versions have been produced independently of Sinclair, e.g. the MGUK ROM by John Alexander, and the JS 4MB version of the JS ROM for use with QL emulators which may have larger memories than a standard QL could offer.

CST updated the original QL ROM into a version specifically for use with its Thor range of computers. This was called Argos. It included its own windowing system and a number of other enhancements to QDOS.

Ultrasoft made a version of the MG ROM, called Ultra-MG, which contained some bug fixes and expected a German key map.

The Tyche ROM is a 64KB development of QDOS for an unreleased QL system, although it probably cannot be used with a standard QL.

The Sinclair ROMs AH, JM, JS, MG and TB have been released for free use in Europe (Sinclair allow it to be distributed for non-profit making use), although rights to these ROMs are held by Paul Holmgren and Frank Davis for North America and permission should be sought from them before downloading them in or for use in North America. The Minerva ROMs are OK to use with emulators etc worldwide.

BASIC Version	QDOS Version	Description
FB	1.00	Sinclair FB ROM (original "kludge" release)
PM	1.01	Sinclair PM ROM
AH	1.02	Sinclair AH ROM
JM	1.03	Sinclair JM ROM
TB	1.03	Intermediate Sinclair QL ROM, between JM and JS
JS	1.10	Sinclair JS ROM
JSU	1u10	Sinclair USA JS ROM
JS-4M	1.10	Sinclair JS ROM, patched for 4MB RAM
MG	1.13	British MG ROM with custom key layout
MF	1.14	A German ROM version
MGE	1E13	Spanish MG ROM version
MGG	1G13	Another German ROM version
MGI	1I13	Italian MG ROM version
MGUK	1ê13	John Alexander updated MG ROM with added commands
JSL1	1.61	Very early version of Minerva ROM
JSL1	1.63	Slightly later version of Minerva ROM
JSL1	1.64	Slightly later version of Minerva ROM
JSL1	1.66	Slightly later version of Minerva ROM
JSL1	1.89	Version of Minerva released for use with QL emulators,
JSL1	1.98	More recent version of Minerva Mk1, freely distributable
Thor 630	1.13	Thor v6.30 ROM
Thor 634	1.13	Thor v6.34 ROM
Thor 636	1.13	Thor v6.36 ROM
thor 639	1.13	Thor v6.39 ROM
Thor 641	?	Thor v6.41 ROM
Tyche	2.05	Last unreleased Sinclair OS for QDOS hardware. A 64K ROM which is interesting but not very compatible with 48K QL ROMs, containing the copyright message '(C) 1985 Sirius Cybernetics'
Ultra-MG	1.14	Ultrasoft release of MG ROM, with some bug fixes. Expects a German keymap.
EFP	1E13	Sigma FP Greek ROM, Version EFP (N.B. E=sigma character, which I don't know how to replicate here)

ROM versions prior to JS (e.g. AH, TB, JM) were early Sinclair releases. They are perfectly usable and many people still have AH EPROMs or JM ROM QL systems.

Version JS introduced several new features such as WHEN ERROR error trapping, WHEN variable value monitoring, and TRA translate features. JSU was a version of the QL ROM for North America.

With the MG ROM came several international versions such as MGF for France, MGI for Italy and so on. Over the years, there have been some derivatives made, such as the 4MB version of the JS ROM, and John Alexander's MGUK ROM.

It's not really known for certain how the two-letter Sinclair ROM versions came to being, one theory had it that they were initials of Sinclair staff (e.g. JM may have been the initials of engineer John Mathieson), or even taxi drivers who ferried QL bits and pieces around for Sinclair! There were even rumours that AH stood for "Angela's Holiday", although after all these years I still can't confirm that!

A bit of a novelty is the Tyche ROM, a final and unreleased 64K ROM for QL hardware. Apparently it can be persuaded to run on the QL emulator QLay, according to Phoebus Dokos.

ROM Bugs

Over the years, it became apparent that various bugs existed in the various ROM versions. Writers such as Simon Goodwin and Mark Knight have documented these. Mark Knight's list is available on my website at:
www.dilwyn.uk6.net/basic/index.html

and Simon Goodwin's articles may be found in back issues of QL World magazine as follows:

- August 1987, page 18, "Bugging The ROM" (this article also contains details of how to convert a machine with an early EPROM version to a later version of QDOS on ROM)
- September 1987, page 12, "Beating The Bugs"
- June 1988, page 30, "Return Of The ROMs"
- February 1989, page 18, "Bugs At Large"

Many of the bugs documented are not serious and have simple workarounds, while some are potentially serious. However, most if not all of these bugs can be avoided by compiling your SuperBASIC programs with Turbo or QLiberator compilers, both of which do a good job of fixing most bugs. The Minerva ROM has also addresses many if not most of the issues with the earlier Sinclair ROMs.

Do bear in mind that although such long bug lists may affect your confidence in using the QL, many of them have minimal effect and the list is probably no worse than the average home computer, it's just that they are more openly documented on the QL.

Printing with SMSQ/E and PCs

by Jochen Merz

Most readers who own QPC, especially QPCPrint owners, should be happy with the current printing situation.
If they still own a printer which "understands" EPSON or PCL printing languages and their programs are written to send their print output in the correct language, everything is fine. Well, at least as long as they don't change their PC system.
And if they own one or more new printers which don't understand a printer language anymore, but come with proper Windows drivers, then QPCPrint will do the job and convert any ESC/P output to something all windows printers will understand.
Hmmm, so, where is the problem? As I mentioned above, if you have an old(er) printer which connects to you system via parallel port and/or serial port, and upgrade your PC system, you may find that you have a problem: there is no serial port plug anymore (well, if you're lucky, there is ONE serial port left), but on all modern PCs I've seen so far, the more important parallel port is history. No fun sending large printouts through a serial port, "even" if it runs at 19200 or 38400 baud.
I have a similar problem, as I have a nice EPSON POS printer (a small slip printer which is used for printing on envelopes or credit card slips), which does not understand ESC/P, and which has no USB port either. Same problem: my old EPSON laserprinter has no USB port, and when I use it, I have to use it in "GQ-Mode" (EPSON page printer mode, history as well ... but the only way to print most QL manuals. The automatic paper feed is broken and nobody seems to be able to repair it - at least not at an affordable price - so I need to feed all sheets manually, and I have to wait until the printer beeps for the next page, otherwise the manual feed will not work properly either ... oh dear!).
Anyway, I need to be able to connect these printers, and I need to send them data WITHOUT QPCPrint or any other Windows driver getting in the way, modifying (i.e. deleting) the data. AND it would be nice to have them installed as "real" printers, so that I can access them from other QPCs over the network.
Sounds somewhat impossible, doesn't it?

The first task is the physical connection. As current PCs come with 4, 6, or 8 USB connectors, it should be no problem to find USB to serial or USB to parallel port converters. And if all USB plugs are already in use, a USB hub could help.
Here a few tips from my experience, which have cost me a lot of nerves (and time and money):

1. If you buy this kind of adaptors, make 100% sure that they are supported well (i.e. that they have XP drivers or, if you run under Vista, Vista drivers). If it does not explicitly state that it is supported by your operating system, DON'T BUY IT! Avoid Xircom, Entrega, IBM.

2. Avoid multi-function-hubs and converters, like Prolific-based hubs with SER, PAR, LAN, USB hub all in one etc. If one driver fails, you have a problem.

3. Do not connect the adaptors into USB hubs if you do not have enough plugs on your PC left. Some need a lot of power, and the parallel port requires power as well.

4. A USB to parallel port bridge is not a full EPP/ECP parallel printer port. This can have several implications. Some converters just do the minimum, i.e. send data to the printer. Don't expect that all the control lines are monitored, and don't use it for anything else (EPROM programming etc.)

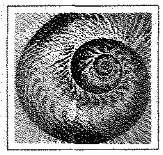
OK, let's assume you have bought a USB to SERial or USB to PARallel converter, plugged it into your computer and installed the drivers.
Then it is time to "find" it in your system. You should see it under "Connectors (COM and LPT)". "Should" is probably "will" in case of a serial "COM" connector, if the installation was successful. You should install the USB to SER connector as soon as you (re-)install your Windows system as possible, as Windows counts every serial port ever installed up by one. Say, you plug in a Bluetooth dongle, and you get two Bluetooth Communication ports (COM1, COM2). Plug the same dongle into another USB plug, you'll get COM3, COM4 ... and so on. This can apply to other COM devices (phones with USB or serial cables etc.) so you may find yourself with two-digit COM ports. Luckily, recent versions of QPC allow you to assign the QL SERial port numbers 1 to 8 to any COM port up to 32. The USB converter I recently installed had a port number of COM31 ... that's close to the limit. So, best install it a.s.a.p. to get a low COM port number. I wonder if there is a way to re-set the port number to get rid of unused old port numbers. Some Bluetooth devices allow you to change the port number in the low range (1 to 8), but that's not really helpful in my case.
Anyway, I knew it was COM31. Fine.
Worse with the parallel port converter. I never found an LPT1 under the COM and LPT settings. I thought the installation failed, and the manual did not explain that the installed device was registered as one of the many USB printer support devices under the USB controller tab. Great! One USB adaptor is listed under the output port category, the other under the input. Doesn't make much sense to me, but so be it. Unfortunately, the USB printer support was not numbered, and there were many (for my many USB printers!) Rightclick, Properties, and you get an idea. And, the last one in the list was the latest one added. Don't know if this logic always works.

After I had identified the right connectors, it was clear that I had to install the printers as Windows printers somehow. OK, I could define SER1 to be COM31, but that would only give me access from local QPCs. But what about the parallel port not being LPT1 ... 8? It had to be a printer. But what kind of printer?
Create a new printer, OK. No plug and play for the port ... of course, not! Uh, and the next window allowed me to choose the right port - there was COM31 apart from many other COMs and, numbered virtual USB printer ports. Luckily, the virtual printer port with the highest number was a real one, not just virtual, as I discovered later. OK, click on "Next".
And now the real problem: which printer to select? There was no "raw data" or something similar. I went through all the manufacturers I knew, but I thought that all the printer models listed there would be driven through a Windows graphics driver in graphics mode, and I was right when I did some tests. Even selecting old EPSON models did not help, and there was no tab in the printer properties to specify "raw data".
... hmmm ... well, there was a printer manufacturer called "Standard". Of course, not a manufacturer, it was a group of Standard printers. Would you expect "Generic / Text only" to be standard nowadays? Or for the last 10 years? Definitely NOT. But that's exactly what I needed - it does not change, modify or translate anything.
So whenever you want to send unchanged, raw data to your printer, select "Generic / Text only".
The rest is the same as for "normal" printers, to make it available on the network etc.
All you need to ensure is, that you do NOT click on the "Use filter" (use QPCPrint) box for these printers in the SER/PAR configuration menu of QPC.
Both printers "sort of" work with the two adaptors. Not 100% perfect, but what do you expect...?
The POS printer on the serial port adaptor now has the problem that, whenever I turn the PC off, I need to turn that off first, otherwise it goes haywire and prints masses of rubbish (probably undefined state of one of the handshake lines).
The old EPSON Laser now has the problem, that it first prints a rubbish page containing an "E" only, when I turn it on. That's technological progress ... or maybe blame it on the Quantum Theory, as there may be other working printers somewhere in this Universe...

Do you remember...? – Part 5

by Ralf Rekondt

QL Technical Guide



For serious programmers, this detailed 195 page manual is probably the most reliable documentation of the QL's firmware which is available.

Written by David Karlin, QL hardware designer, and Tony Tebby, designer of Qdos, the QL Technical Guide takes the user into many aspects of the QL which have previously not been documented in such detail. The information it gives will be useful to anyone who wants to explore the QL in greater depth than the QL User Guide covers and especially to machine code programmers.

- Subjects covered in detail include full documentation of the QL's memory allocation, the complete specification of Qdos system calls, and the layout of Qdos system variables. There is also information on interfacing machine code programs to SuperBASIC.
- Other chapters of the guide cover ways in which various peripherals such as hard disk interfaces, add-on memory, and ROM cartridges may be added to the QL. These sections include many details about how the firmware for such devices should be written.
- Finally, for users with a commercial interest in programming, there are details of Sinclair's options for the distribution of QL Software and the purchase and duplication of Microdrive cartridges.

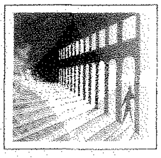
Price:
£14.95

written by:
**David Karlin
and
Tony Tebby**

6509

sinclair

QL Project Planner



Any complex task can waste time and money if things aren't ready when they are needed. But your QL can help you to plan the most efficient schedule.

Any business needs to make its long-term plans as efficiently as possible, but even if you don't have a company to run, you may have to organise things like moving house or building an extension. The problem that all such tasks introduce is how to plan a project that involves several activities over a long period, many of which depend on the others being complete so they can start.

- QL Project Planner is designed to deal with the problem and produce the most efficient schedule by helping you to think about the project systematically. You start by breaking the project into component activities, assigning each one a duration, and saying which others it depends on.
- Using the techniques of Critical Path Analysis, QL Project Planner builds up a chart which shows each activity in relation to the others. You can see which cannot be altered without affecting the whole project, and which are flexible. When all the activities have been listed you can produce a working schedule.
- Everything is designed to be easy to follow on screen, and there is also a comprehensive teaching program. For a look at how QL Project Planner works, see the other side of this sheet.

FINALIST
MICRO
BRITISH COMPUTING
AWARDS
1985

Price:
£39.95
(includes comprehensive manual)

BRAINPOWER

Software written by:
Triptych Publishing Ltd.

5505

sinclair

With this part, we have reached the end of the series - at least we've used up all the material we have received from Ralf. Enjoy.

QL Technical Guide

(to the left). In the early days THE reference manual for everybody who wants to know (most things) about QDOS. This manual was later replacted by the QDOS/SMS Reference manual, which corrected some misunderstandings in the QL Technical Guide, corrected a few bugs but, most important, added loads of useful information which was missing in the QL Technical or had been updated. As SMSQ/E got more and more features, it was clear after some years, that a printed documentation was unable to keep in step with the updates - and nowadays, thanks to the Internet, PDF and Dilwyn Jones, most information is available in electronic format.

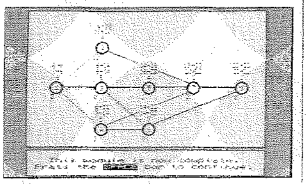
QL Project Planner

(below) Requires SPK and BASIC. Program to create projects - has anybody ever used it?

Using QL-Project Planner

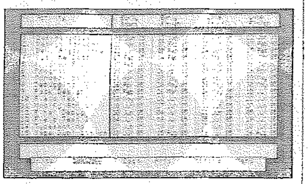
Making a Critical Path Analysis with paper and pencil is messy and time-consuming. The QL version is neither, since the computer performs all the hard parts for you.

The normal way to plan a complex task with paper and pencil involves preparing a diagram showing the relationship of all the stages of the project. Because this can be difficult to work out, it usually involves many corrections and additions.



The QL Project Planner teaching program demonstrates what is involved in making such a diagram by hand. *You* won't need to prepare a chart like this, but it makes it easier to understand the theory.

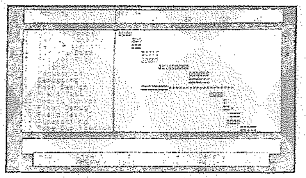
Each circle on the screen represents one of the project's component activities. The diagram shows the duration of each one, and which activities are 'critical'. This means that



they cannot start later (or finish earlier) than scheduled, without affecting the length of the whole project. These are the activities which must run to time, and the chart also shows those outside the 'critical path', where there is some leeway.

QL Project Planner cuts out all the complication of planning a project in this way, by performing the calculations for you. You just break down the project into a simple table, specifying the various activities and a duration for each of them, then state which other activities they depend on. As soon as you have done this, QL Project Planner can calculate and display the critical path.

If you want, you can specify the costs of each stage as well. You can amend all the details of the list as often as you like until you are satisfied with the data.



QL Project Planner makes it easy to understand the results. You can request a bar chart on screen or printer, showing you how to organize the project schedule. This distinguishes non-critical activities and indicates the margins within which they can move without becoming critical.

The same information can be given in a calendar form if you prefer. QL Project Planner also makes it possible to extract information about things like the cost of the project, and because it is so easy to experiment, you will be able to see whether you can make any changes to make the project run more efficiently.

RWAP Services NEWS!

**** We have moved ****

See our updated address details below.

We have also acquired more brand new Sinclair QL membranes and another stock of Epson Stylus Colour 850 inkjet printers, so if you need a better printer for your QL, give us a shout.

More news is always available on our website: www.rwapsoftware.co.uk

We are also looking to produce some new hard disk interfaces for the ZX Spectrum and have a few little projects on the drawing board.

Our websites:
http://www.rwapservices.co.uk (General site)
http://www.rwapsoftware.co.uk (Sinclair computer second hand and new items)
http://www.rwapadventures.com (Adventure Programs)
http://www.internetbusinessangels.com (Guidance on setting up online businesses).

New Products!

QWORD 1.0 NOW WITH DIGITAL SOUND ON QPC2!

The wait is now over! Q-Word version 1 is finally available!

Platforms:
QPC/QXL, Q40/Q60, Aurora (with SGC)

Prices:

All versions without P-Word	£20.00
All versions with P-Word	£30.00

Notes:
Q-Word DOES NOT require SMSQ/E with GD2 support -OR- SMSQ/E at all on the Aurora or Qx0 machines. It works on the highest colour depth everywhere regardless of Operating System.
The Aurora version is available on either HD or ED disk. For the latter add £1.00 to the price. ED version is uncompressed and can be run directly from the floppy. All other Floppy versions are compressed. QPC/QXL version comes on CD. Non CD versions DO NOW support digital sound on QPC2

TALENT COMPUTER SYSTEMS for **Windows**

For QLers that run Windows or with incompatible hardware for Talent Games, we now have re-released these adventures so that they can run on your Windows-equipped PC. No Emulator, floppies, microdrive backups etc. required, just a one-click install! Of course the full QL line is still available! (See side column)

Talent Games for Windows	ea. £ 10.00

(Each Game includes a runtime installation of QLAY-2 by Jimmy Montesinos)

Games Currently Available from www.rwapadventures.com

The Lost Kingdom of Zkul
West
The Prawn
Return to Eden

Replacement Sinclair QL Keyboard Membranes

We always have a stock of brand new Keyboard Membranes (and keyboard parts) for the original Sinclair QL, so if you have some keys which no longer work, just give us a call.

Cost is only £18.50 plus £2.75 post and packing.

Second Hand Items - Huge Range Available

We stock a wide range of books, hardware and software for the Sinclair QL, Z88 and ZX Spectrum, including disk interfaces, memory expansion and microdrive cartridges. If there is anything you need - have a look at www.rwapsoftware.co.uk (or ring us with details of your requirements).

We are always happy to help.

Old Favourites!

Utilities

SBASIC / SuperBASIC Reference Manual on CD	£ 20.00
Sidewriter v1.08 *Landscape Printing (EPSON printers)*	£ 10.00
ImageD v1.03 *3D object generator*	£ 10.00
Q-Help v1.06 *Superbasic On-Screen help system*	£ 10.00
Q-Index v1.05 *Keyword-to-topic finder*	£ 5.00
ProForma ESC/P2 Drivers v1.04 for ProWeSs *Printer Driver*	£ 8.00

Applications

Flashback SE v2.03 (upgrade only) *Database*	£ 5.00
QL Cash Trader v3.7 *Accounting/Finance*	£ 5.00
QL Payroll v3.5 *Accounting/Finance*	£ 5.00
QL Genealogist v3.26 *Genealogy*	£ 20.00
Genealogy for Windows	£ 50.00
QL Genealogist to Windows version upgrade	£ 25.00
QL Cosmos v2.04 *Planetarium*	£ 5.00
Q-Route v2.00 *Route Finding*	£ 25.00
Upgrade from v1.xx	£ 5.00
Britain map v1.11	£ 2.00
BIG Britain map (needs 2Mb) v2.03	£ 5.00
Various Britain Area maps (ask for details)	ea. £ 2.00
Ireland map v1.00	£ 5.00
Belgium map v1.01	£ 2.00
Catalonia map v1.03	£ 2.00
P-Word UK English Dictionary (500.000 words!) *Dictionary*	£ 15.00

Leisure

Return to Eden v3.08 *Adventure*	£ 10.00
Nemesis MkII v2.03 *Adventure*	£ 8.00
The Prawn v2.01 *Adventure*	£ 8.00
Horrorday v3.1 *Adventure*	£ 8.00
West v2.00 *Adventure*	£ 5.00
The Lost Kingdom of Zkul v2.01 *Adventure*	£ 5.00
All 6 games above	£ 25.00
D-Day MkII v3.04 *Strategy/War Simulation*	£ 10.00
Grey Wolf v1.08 *Graphical Submarine Simulation*	£ 8.00
War in the East MkII v1.24 (upgrade only) *Strategy/War Simulation*	£ 5.00
Open Golf v5.20 *Sports Simulation*	£ 8.00
QuizMaster II v2.07 *Quiz*	£ 5.00
Stone Raider II v2.00 *Arcade Game*	£ 5.00
Hoverzone v1.2 *Arcade Game*	£ 5.00
Deathstrike v1.5 *Arcade Game*	£ 5.00
Flightdeck v1.0 *Flight Simulation*	£ 10.00
All 6 games above (Open Golf, QuizMaster II,Stone Raider II, Hoverzone, Deathstrike and Flightdeck)	£ 28.00

Notes on Software requirements
The following programs have a minimum SGC card requirement: P-Word, Qword, Big Britain MAP for Q-Route

RWAP Services
3 Dale View Court, Fulford, Stoke-On-Trent, Staffordshire ST11 9BA TEL: (+44) 1782 398143 From the UK Dial: 01782 398143
Website: http://www.rwapsoftware.co.uk
Email: sales@rwapsoftware.co.uk

We Accept Payment using:

PayPal VERIFIED NOCHEX EMAIL MONEY PREFERRED PAYMENT (Cheques in £ sterling made payable to R. Mellor)

QPAC2 Button Frame

by Dilwyn Jones

The button frame is an ingenious little facility built into QPAC2 which amounts to a list of "sleeping" programs. Many pointer driven programs have a "sleep" button which reduces the program display by minimising all open windows so that the only visible part of the program is a little button in the button frame which displays the name of that program. Pressing the right mouse button ("DO") or pressing the ENTER key while the pointer is over the button wakes up the program again. Normally, only programs which have a sleep icon (usually shown as a 'Zzz' icon in QDOS programs, or a downward pointing arrow in more recent SMSQ/E programs) can be zapped into the button frame.

But, with a simple hotkey definition, any QPAC2 user can quickly and simply zap even older programs like Quill into the button frame!

We all know that many programs have a "sleep" button which puts the program to sleep in the QPAC2 button frame. But many programs don't.

I, for one, make use of nice colour background wallpaper via the BGIMAGE command. Well, I did compile the Wallpaper CD and DVD, so after all that hard work preparing a collection of pictures I may as well use it.

I'm also a fan of multitasking and task switching, so I tend to have a lot of programs running in memory at the same time, which leads to a cluttered display and of course hidden wallpaper.

So, any program I'm not using at the time gets to sit in the button frame until I need to use it. As an example, I have Quill running, into which I'm typing an article for QL Today, Archivers Control Panel is still in memory, along with Launchpad, Abacus and a few other little tools. Launchpad is OK, it has its own sleep button. Quill and Abacus don't.

QPAC2 has a handy little "thing" called button sleep. Its purpose is simple - it zaps the current program into the button frame. It's not particularly obvious how to use it, however.

The best way I've found is to set up a hotkey command, so that button sleep can be simply and quickly called from whatever program you're currently using. It works for BASIC too - the main BASIC interpreter gets zapped into a button called "System".

```
ERT HOT_WAKE(CHR$(233),'button_sleep')
```

Now all you have to do is press CTRL ALT F1 (hold down CTRL and ALT keys and tap the F1 key) and voila, the current program gets zapped into the button frame where it stays until you select it again in the usual way for a button.

The two screen dumps show the difference this can make. Figure 1 shows a rather extreme example with an extremely cluttered display and only a tiny amount of the wallpaper visible.

Figure 1

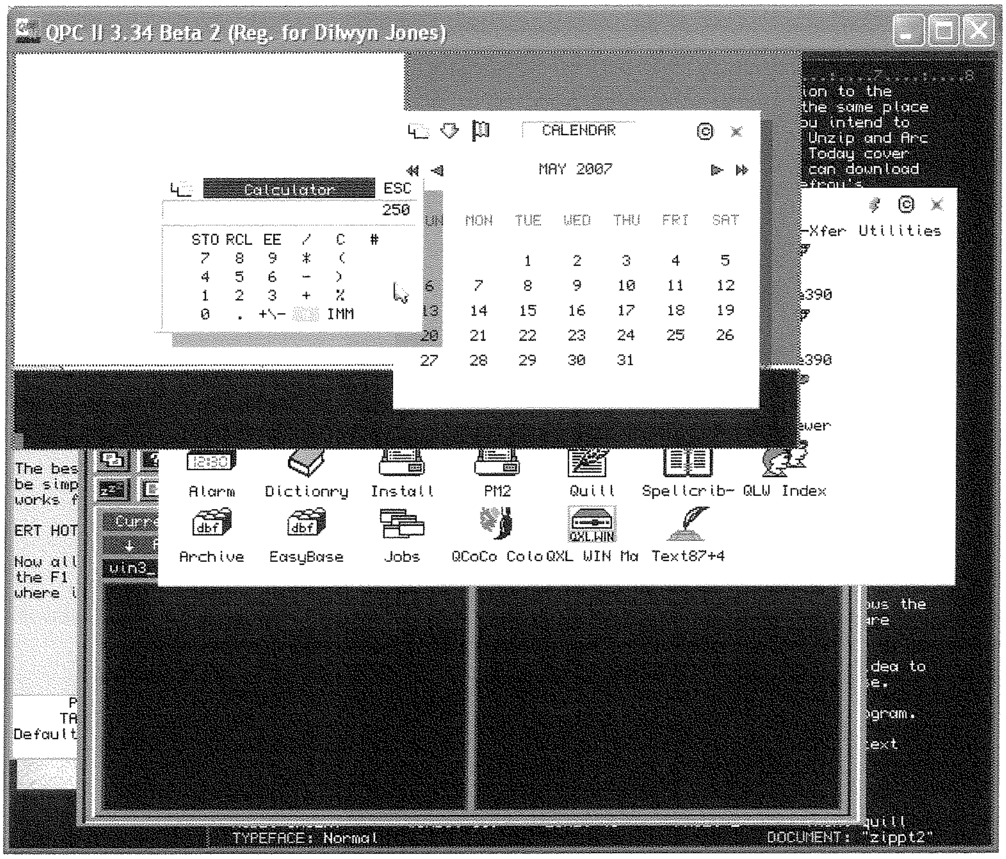


Figure 2 shows what it's like after the rather extreme act of zapping everything into the button frame. Now you can see the nice background picture and you have a row of buttons across the top from where you can just select the next program you wish to work with!

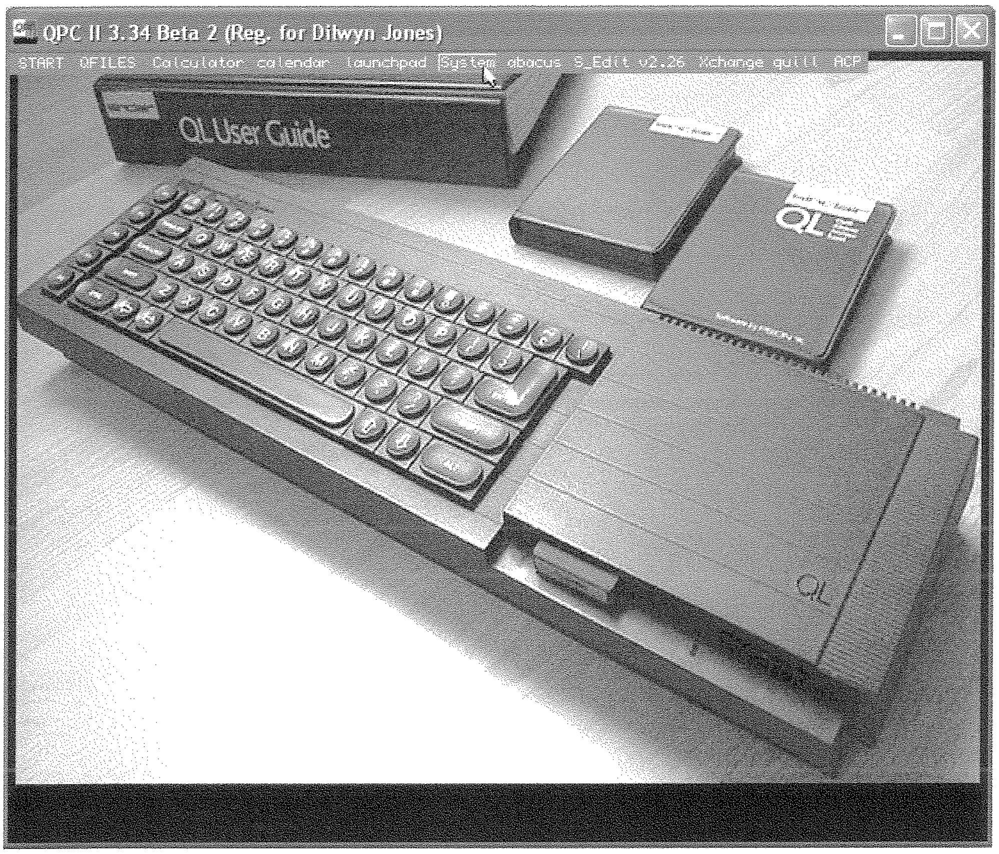


Figure 2

Compare figure 3 with figure 1. Figure 3 is a much less cluttered display with just one program at a time active and everything else neatly in the button frame.

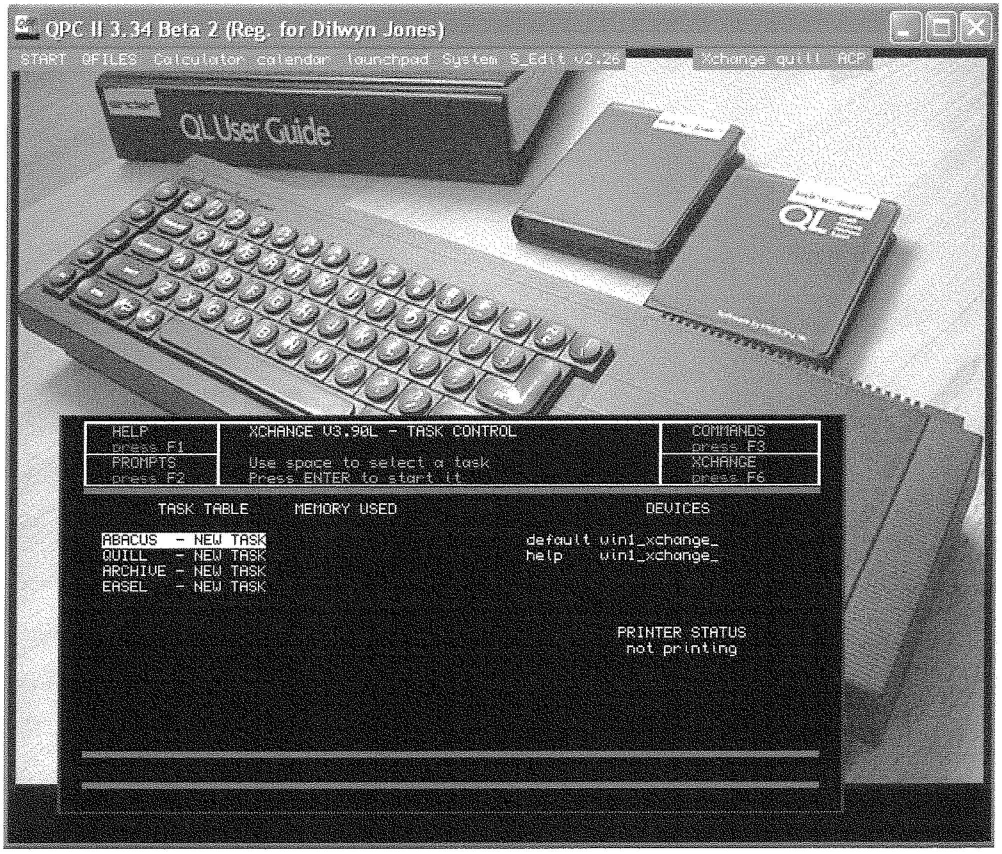
Of course, if everything is in the button frame, you might also want a hotkey to quickly and simply select the button frame itself. This key definition will pick the button frame, making it visible with the pointer in place ready to select something from the buttons list:

```
ERT HOT_WAKE('.','Button_Pick')
```

so that all you have to do is press ALT . (hold down the ALT key and tap the full stop key) to select the button frame.

It is even possible to configure your system so that pressing both buttons at the same time generates the equivalent of ALT "." - this makes it even easier to select the button frame by just pressing both mouse keys at the same time! Configuration of the keypress generated when both mouse buttons are pressed is done in the System part of the QPAC2 configuration - alter the setting of the option called "Hotkey when both mouse buttons are pressed."

Figure 3



```
HELP        XCHANGE V3.90L - TASK CONTROL         COMMANDS
press F1                                          press F3
PROMPTS     Use space to select a task            XCHANGE
press F2    Press ENTER to start it               press F6

     TASK TABLE     MEMORY USED                       DEVICES

ABACUS  - NEW TASK                          default win1_xchange_
QUILL   - NEW TASK                          help    win1_xchange_
ARCHIVE - NEW TASK
EASEL   - NEW TASK

                                                PRINTER STATUS
                                                 not printing
```

Working Space 2

by George Gwilt

I have already written about the use of working space in stand alone executable assembler programs. I discuss here methods to be used in code which is either to be CALLed or which forms part of a machine code procedure or function. The problem here is that the code is operating as part of SuperBASIC.

Certainly on the original QLs, SuperBASIC could be moved at any time, and this means between any two instructions. Given that A6 is the pointer to the SuperBASIC data area this means that A6 may change at any time. So might the stack pointer, A7.

How does this affect the choices of space for storing information? For an executable program the choices were:

- The program itself
- The program's dataspace or stack
- The free memory or heap.

The first choice is no use for machine code routines as that would render them non re-entrant. This leaves the other two. Using the common heap has the danger of heap fragmentation so this leaves dataspace or stack.

However, in SuperBASIC, there are two other areas available. The first is the Basic Buffer and the second is the Maths Stack. To find where these are we look at specific places in the Basic data area.

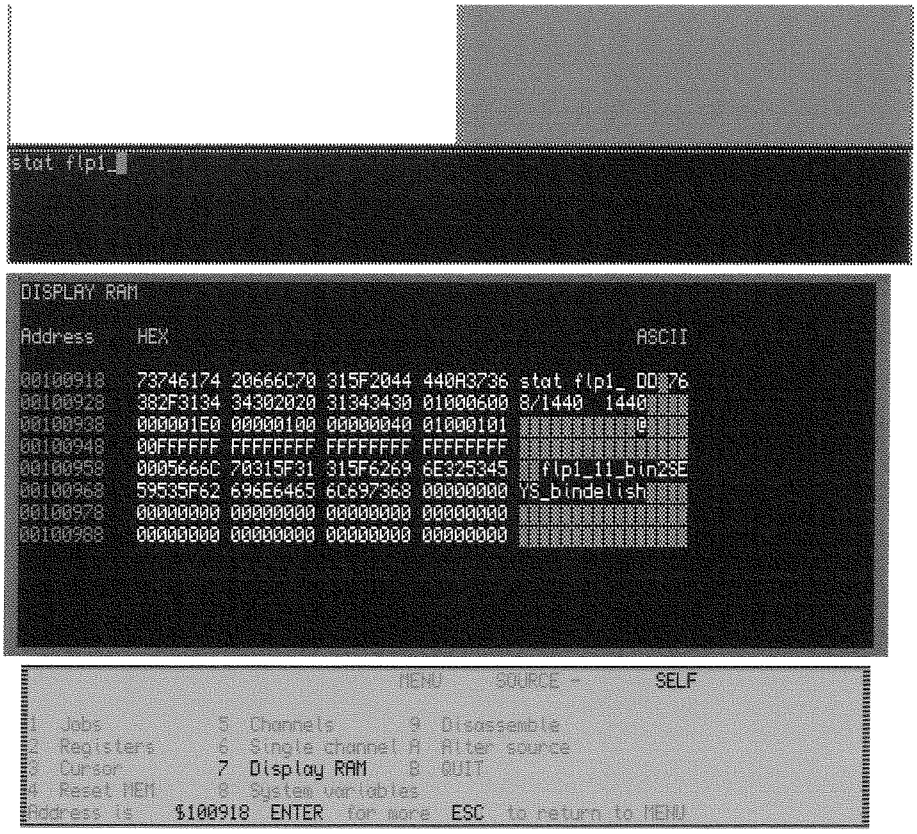
BV_BFBAS(A6) points (relative to A6) to the base of the Basic buffer
BV_TKBAS(A6) points (relative to A6) to the end of the Basic buffer
BV_RIP(A6) points (relative to A6) to the Maths Stack

```
BV_BFBAS =   0
BV_TKBAS =   8
BV_RIP   = $58
```

Basic Buffer

If you type a few letters on the keyboard they will appear in channel 0. If you look at the Basic Buffer you will see the same letters at the start of the buffer. If you type, for example, "stat flp1_" followed by ENTER. you will see the information about flp1_ in channel 1. At the start of the Basic Buffer you will see much the same information about flp1_. This shows the Basic Buffer being used by SuperBASIC. You can use it too.

The following two figures show the contents of the first part of the Basic Buffer when "stat flp1_" has been typed and then after ENTER is pressed.

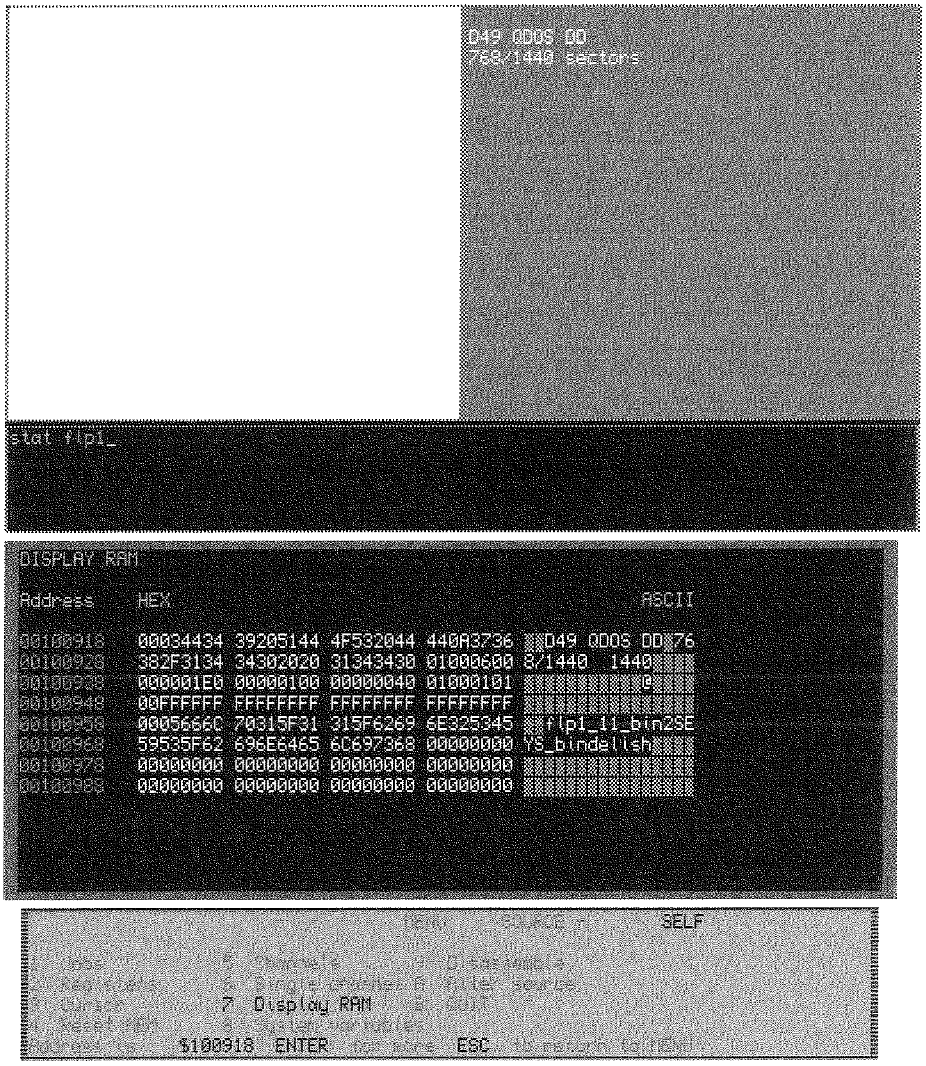


```
stat flp1_
```

```
DISPLAY RAM

Address    HEX                                  ASCII

00100918   73746174 20666C70 315F2044 440A3736 stat flp1_ DD▒76
00100928   382F3134 34302020 31343430 01000600 8/1440  1440▒▒▒
00100938   000001E0 00000100 00000040 01000101 ▒▒▒▒▒▒▒▒▒▒▒▒@▒▒▒
00100948   00FFFFFF FFFFFFFF FFFFFFFF FFFFFFFF ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
00100958   0005666C 70315F31 315F6269 6E325345 ▒▒flp1_11_bin2SE
00100968   59535F62 696E6465 6C697368 00000000 YS_bindelish▒▒▒▒
00100978   00000000 00000000 00000000 00000000 ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
00100988   00000000 00000000 00000000 00000000 ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
```



```
D49 QDOS DD
768/1440 sectors

stat flp1_

DISPLAY RAM

Address    HEX                                          ASCII

00100918   00034434 39205144 4F532044 440A3736  ▒D49 QDOS DD▒76
00100928   382F3134 34302020 31343430 01000600 8/1440  1440▒▒▒
00100938   000001E0 00000100 00000040 01000101 ▒▒▒▒▒▒▒▒▒▒▒@▒▒▒
00100948   00FFFFFF FFFFFFFF FFFFFFFF FFFFFFFF ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
00100958   0005666C 70315F31 315F6269 6E325345 ▒▒flp1_11_bin2SE
00100968   59535F62 696E6465 6C697368 00000000 YS_bindelish▒▒▒
00100978   00000000 00000000 00000000 00000000 ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
00100988   00000000 00000000 00000000 00000000 ▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
```

The size of the buffer is found from
BV_TKBAS - BV_BFBAS.

In SMSQ/E this defaults to 1024 bytes.

Maths Stack

The Maths Stack is used, obviously, for maths operations but also for storing parameters of machine code routines. This stack, like the user stack, is "upside down". That is, when an item is pushed onto the stack the stack pointer points to a lower address after the operation.
To use space on the Maths Stack you should first see that there is room, by using the vector BV_CHRIX. Then you should reduce the address in BV_RIP(A6) by the amount of space you want. Simple!

Example

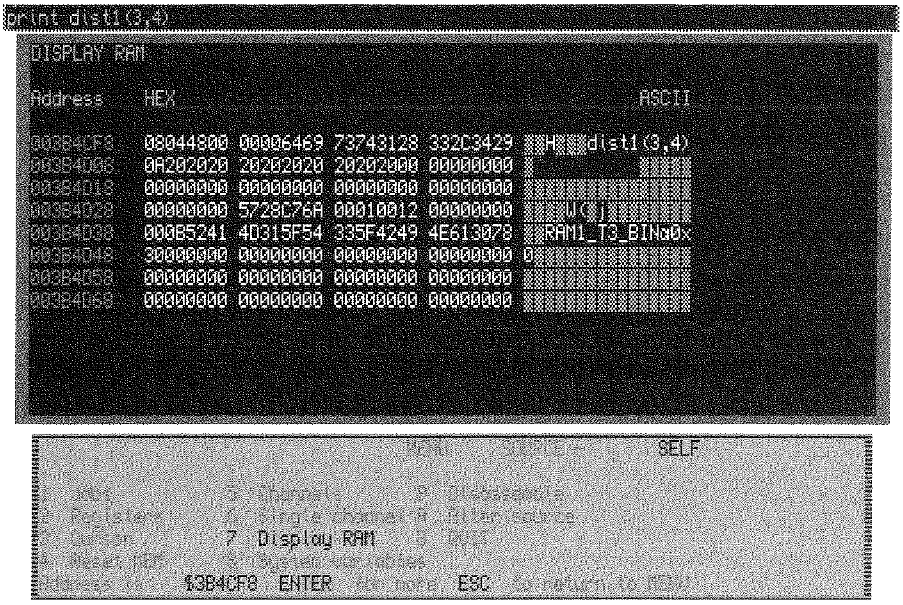
I will give an example now to illustrate these.
Suppose we want to write a machine function finding the distance from the origin to the point x,y. This distance is (x^2 + y^2)^.5.
To show the different ways of using storage I will write code for two functions, DIST1 and DIST. DIST1 will use the Basic Buffer and DIST will use the Maths Stack.

The two parameters to DIST and DIST1 are, of course, x and y. Thus
DIST(3,4) and **DIST1(3,4)**
should both give us the answer **5**

The calculations, performed by RI_EXECB, are given in SET, for DIST, and SET1, for DIST1. These are identical except that SET has an extra $FB at the end.

This is what happens:

Op code	TOS	NOS	Next	Storage	
	x	y	–	–	
QA.DUP	x	x	y	–	
QA.MUL	x^2	y	–	–	
$FB	y	–	–	x^2	
QA.DUP	y	y	–	x^2	
QA.MUL	y^2	–	–	x^2	
$FA	x^2	y^2	–	x^2	
QA.ADD	x^2+y^2	–	–	x^2	
QA.SQRT	(x^2+y^2)^.5	–	–	x^2	
$FB	(x^2+y^2)^.5	–	–	(x^2+y^2)^.5	(Only SET)



```
print dist1(3,4)
DISPLAY RAM

Address    HEX                                   ASCII

003B4CF8   08044800 00006469 73743128 332C3429   H  dist1(3,4)
003B4D08   0A202020 20202020 20202000 00000000
003B4D18   00000000 00000000 00000000 00000000
003B4D28   00000000 5728C76A 00010012 00000000      W(j
003B4D38   000B5241 4D315F54 335F4249 4E613078   RAM1_T3_BINa0x
003B4D48   30000000 00000000 00000000 00000000  0
003B4D58   00000000 00000000 00000000 00000000
003B4D68   00000000 00000000 00000000 00000000

                          MENU    SOURCE -        SELF

1  Jobs             5  Channels         9  Disassemble
2  Registers        6  Single channel A  Alter source
3  Cursor           7  Display RAM      B  QUIT
4  Reset MEM        8  System variables
Address is     $3B4CF8  ENTER  for more  ESC  to return to MENU
```

As you will see from the code which follows, the extra $FB in SET copies the answer to the place on the Maths Stack which is to hold the result for the return.

The code is given later but first I show the contents of the Basic Buffer just after the call to RI_EXECB in DIST1.

This shows that the fp value:

080448000000

is stored at the start of the buffer. Shown also in channel 0, is the command which started this off:

print dist1(3,4)

The fp number is 9 which is what we expect since it is x, the first number on the stack, which is squared and stored.

CODE for DIST1 and DIST

```
; Both DIST1 and DIST do the same thing which is to find the distance
; from the origin to x,y.

; DIST1(x,y) and DIST(x,y) both return (x^2+y^2)^.5

; The difference is in the space used for storing intermediate results.
; In both cases the Maths Stack contains x,y after the parameters are read.
; In both cases RI_EXECB is used to perform a set of operations. In the
; case of DIST an extra final operation sets the answer in the storage
; space, which is on the Maths Stack at its original place.

; For DIST1 we use the Basic Buffer. First we check that there are six
; bytes available (sensible for rather larger numbers). Then we set
; A4 pointing to the end of our space.


DIST1     MOVE.W     DIFF,D6              ; Mark DIST1
          MOVEA.L    BV_BFBAS(A6),A4      ; Pointer to Basic Buffer's start
          LEA        6(A4),A4             ; Add the 6 bytes needed
          CMPA.L     BV_TKBAS(A6),A4      ; Is this too big? . .
          BHI        BUFFUL    ——›        ; . . yes!
          BRA        D_1

; For DIST we use part of the Maths Stack. First we check that there is
; space for our 6 bytes. Then we set A4 pointing to the end of the 6 bytes.

DIST      MOVEQ      #0,D6                ; Mark DIST
          MOVEQ      #6,D1                ; See that 6 bytes are . .
          MOVEA.W    BV_CHRIX,A2          ; . . available on the . .
          JSR        (A2)                 ; . . Maths Stack
          MOVE.L     BV_RIP(A6),A4        ; Address of our space
          SUBQ.L     #6,BV_RIP(A6)        ; Reserve it
```

```
D_1       MOVEA.W   CA_GTFP,A2         ; Arrange to get FP parameters
          JSR       (A2)
          BNE       BAD_PARM  ----›    ; Something went wrong
          SUBQ.W    #2,D3              ; Are there 2 parameters? . .
          BNE       BAD_PARM  ----›    ; . . no!
          LEA       SET(D6.W),A3       ; Set the operations for DIST1 or DIST
          MOVEQ     #0,D7              ; Just in case
          MOVEA.W   RI_EXECB,A2        ; Do the operations
          JSR       (A2)
          BNE       BAD_EXIT  ----›    ; An error!
          MOVE.L    A1,BV_RIP(A6)      ; Set the Maths Stack for the answer
          MOVEQ     #2,D4              ; Signal FP
          MOVEQ     #0,D0
          RTS


BAD_PARM  MOVEQ     #-15,D0
BAD_EXIT  RTS
BUFFUL    MOVEQ     #-5,D0
          BRA       BAD_EXIT

DIFF      DC.W      SET1-SET

SET       DC.B      QA.DUP,QA.MUL,$FB,QA.DUP,QA.MUL,$FA,QA.ADD,QA.SQRT,$FB,0
SET1      DC.B      QA.DUP,QA.MUL,$FB,QA.DUP,QA.MUL,$FA,QA.ADD,QA.SQRT,0
```

Mice and Ergonomy

by Steve Poole

Ergonomy is the study of the Science of Work, mainly used in company's method's departments to simplify tasks. The mouse was invented in the mid-sixties at the Augmented Human Intellect Research Center, (at Stanford University), by Douglas C Engelbart and was made popular on the MacIntosh personal computer.

When I was a student in the late '60s, we were taught to analyse all of the hundreds of jobs done in the Factory in which I did my sandwich course, and define simple practices common to all jobs. This work was part of the work-study department, whose aim was to increase productivity by eliminating unnecessary effort.

Now that I have sufficient time, I often reflect on the way modern day computerised technology is going. It is interesting to look at what we are being subjected to by the various gadgets offered to the average man by the market, and their typical quality/price ratio tags:

Portable PCs: 900 EUR / Photo printers: 150 EUR / MP3 Walkman: 150 EUR / Scanners: 100 EUR, / Photoframes: 200 EUR / Mobile phones: 300 EUR / Audio kits: 250 EUR / Flat screens: 200 EUR / Digital cameras: 300 EUR / Camescopes: 350 EUR / Innumerable telecommands: 50 EUR each / Modems: 50 EUR / Mains networks: 150 EUR / Harddrives: 250 EUR / GPS navigators: 300 EUR / Pocket audio-video players: 1000 EUR / USB keys: 50 EUR / 5/1 speakers: 150 EUR / TNT cards: 200 EUR / Graphics cards: 150 EUR / Webcams: 30 EUR / Hubs: 150 EUR / Hands-free headsets: 130 EUR / Graphics tablets: 100 EUR / Bluetooth: 60 EUR / Flat-TV: 1000 EUR / DVD portable: 250 EUR / Cordless mouse & keyboard: 70 EUR / Power back-up: 100 EUR.

This list is by no means exhaustive and assumes you are the only member of your family keeping up with technological progress. So far this lot has set you back some 6500 euros, and that is without having bought your desktop, or consumables!

Now what amuses me with this, is that when I buy one of them, I first have to spend several days reading and re-reading the instructions before I dare start setting anything up. If I am lucky, the instructions will be in English. As I live in France, the instructions will probably be in French, but as I understand English I can have a good laugh comparing both texts. For example, in a recent booklet the names Bill and Mr Day are translated as 'Facture' and 'Journée', the former of which means 'invoice'! And I dare not mention all the other howlers... The trouble is that manufacturers expect 'o'-level students to be well-versed in tech-speak, which they are not, and hence translations are nowadays done for peanuts, with the inevitable results. And the trouble with instructions are that they are written by scientifically-orientated engineers who dropped English literature at fifteen, and think they are addressing an audience of electronics technicians. Writing Instructions leaflets is a qualified job in itself, and manufacturers should pay specialists to do the work for them, that is by generalist technicians with NO PRIOR KNOWLEDGE of the product. This would demand some liaising between manufacturers and authors, but clarifications are invariably necessary if the final result is to be comprehensiable.

But before you can write the instructions, you must first understand how the gadget works, that is, these days, how the menu systems have been designed and programmed. This brings us back to Dilwyn's favourite pet: The humble mouse. And I say 'humble' mouse because, like the proverbial scape-goat, we tend to pile all PC faults onto the poor beast's back. And this is where we get bogged down in nightmares: Even on Windows-only software, there is NO consistency in the ergonomy of mouse-interfacing. The ONE rule I learnt was this: Left-click 'space', right-click 'enter'. But even this is generally not respected. When programming on my QPC I never use the mouse at all, as this would necessitate programming with the pointer environment, and with good programming practise we can always use the same keys for the same functions. But with Windows, who can tell if you need to click or double-click on a location if you have never been there before? The only advantage of the mouse is that you can move around the screen faster. Full Stop.

Following lightning damage, I recently replaced my 56k modem with broadband, and of course had to install it and set it up, supposedly 'Plug'n'Play'. But I kept getting into dead-ends and had to begin reinstalling it all over again using trial and error to make progress. I borrowed a book from the library to help me, but it was out of date, ditto for the manufacturer's instructions. Even the installation CD was apparently not fully synchronised with it's text, which was clearly in need of updating. (But perhaps there had been some bungling, because my web supplier had recently been taken over...).

Eventually I got onto the web, but found that the net interface had been tweaked. So John Gilpin, the Quanta Editor kept getting blank emails from me, as the REPLY button had changed from double-click to single-click. Once he had emailed me about this I was able to detect the error and use the interface in the revised way. But at 59, my reflexes are well-engrained, so John should be getting yet more blank emails from me!

So here I am with an increasing number of gadgets, most of whose functions I cannot remember, as they are so inconsistant and definitely not logical, a situation made worse because of all the telecommands lying around the house, who multiply the number of input sequences yet again.

In previous editions of QL Today, there has been referred to a 100 euro portable QL. This project was brought to my notice at an Eindhoven Show some two years ago. Investment funds were available and a suitable manufacturer too. But it seems unlikely that this project will get off the ground now because the Massachusetts Institute of Technology have their own project, even though the price has been upped to 150 dollars, and they are geared up to produce hundreds of millions of these machines for the third world market.

So what is the point of this article? Well, a 100-euro portable QL would probably have simple home-grown interfaces such as QDT grafted onto SMSQ/E. So we could at last expect a consistant, logical front-end system on computers world-wide, which could form a new base to be copied by all the innumerable gadgets which will no doubt be produced for uneducated third-world Citizens too. If only we had our new QL we could chuck out all the confusing mess sold as 'intuitve' technology. Ahh!! Dilwyn, open the Window...

Letter-Box

Tony Firshman writes:

Roy Wood (Byts of Wood 11/5) asks why maths geeks get confused between Xmas (Dec 25) and Halloween (Oct 31). I am sure he explains in this issue.

Most of you will know that according to Douglas Adams the world was created as a program to find the answer to six times nine. The 'wrong' answer, after two million years, was 42, of course. A similar maths geek once excitedly approached Douglas Adams and said he had got it. "It all works in base 13". Imagine his disappointment when he got the reply "Oh yes - what a coincidence". I often wonder whether this was a double bluff - just what Douglas Adams might do in real life.

TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets.

Can be used with lights/hifi/monitors—ie a QL monitor can be used as a switch control.

Cost £24

superHermes

A major hardware upgrade for the QL

All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/ keyclick) IBM AT kbd I/F // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software)	**£90** (£92/£93)
IBM AT UK layout Keyboard	**£11** (£13/£15)
Serial mouse	£8 (£8.50/£9)
Capslock/scrollock LED	**£1** (£1.50/£1.50)
Keyboard or mouse lead	**£3** (£3.50/£3.50)
High speed serial (ser3) lead	**£4** (£4.50/£4.50)

Hermes available for £25 (£26/£27) **Working ser1/2 and independent input, debounced keyboard.**

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.

Cost (incl keyboard lead) .... **£53** (£54/£55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thorn-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating system/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+ Hermes) & built in Multibasic.

First upgrade free. Otherwise send £3 (+£5 for manual if requd).
Send disk plus SAE or two IRCs

MKI...£40 (£41/£43) **MKII...£65** (£66/£67)

MINERVA RTC (MKII) + battery for 256 bytes ram. CRASHPROOF clock & I²C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL

A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage - it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second.

Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off **RomDisq** at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq	**£39** (£40/£41)
4mbytes RomDisq	**£65**(£66/£67)
8 mbytes RomDisq	**£98** (£99/£100)
Aurora adaptor	**£3** (£3.50/£4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost .... **£34** (£35/£36)

I2C INTERFACES

Connects to Minerva MKII and any Philips I²C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)

2 amp (for 8 relays, small motors) .... **£40** (£43/£44)

4 amp total (for motors etc) .... **£45** (£48/£50)

Relays (8 3a 12v 2-way mains relays (needs 2a power driver) .... **£25** (£28/£29)

Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required .... **£25** (£27/£28)

Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue outputs (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting .... £30 (£31/£32)

Temp probe (-40°C to +125°C) .... **£10** (£10.50/£11)

Connector for four temp probes .... £10 (£10.50/£11)

Data sheets .... £2 (£2.50/£3)

Control software & manual (for all I/F) .... £2 (£2.50/£3)

QL SPARES

Keyboard membrane	**no longer on sale**
1377 PAL	**£3** (£3.50/£4)
Circuit diagrams	**£3** (£3.50/£4)
68008 cpu or 8049 IPC	**£8** (£8.50/£9)
8301/8302 or JM ROM or serial lead	**£10** (£10.50/£11)
Power supply (sea mail overseas)	**£12** (£19/£23)

Other components (sockets etc) also available

Prices include postage and packing (Airmail where applicable) Prices are: UK (Europe /Rest of world). Payment by cheque drawn on bank with UK address,/postal order or CASH! I can no longer accept card payments as UK only does PDQ transaction. SAE or IRC for full list and details **22 Feb 03**

29 Longfield Road, TRING, Herts, HP23 4DG
Tel: +44 (0) 1442-828254 Fax/BBS: +44 (0) 1442-828255
tony@firshman.co.uk http://www.firshman.co.uk

A Hiccup in Birmingham

by Geoff Wicks

The week started inauspiciously. When I turned on my router at 7.30 on Monday morning something was not quite right. It took me a little time to discover it, but the DSL light was not burning. My telephone line was dead. Goodbye internet!

It had happened at the worst possible QL Today time. We were just starting to put together this issue of QL Today. At this stage of publication there are often regular emails between Jochen and me. Fortunately I had sent Jochen 1Mb of news and editorial the previous day, but there were things in both I had asked him to read in case they had to be reworded. I had also asked him to hold some space at the back of the magazine as I was going to Birmingham to meet the Quanta's West Midlands subgroup. I hoped to write a short piece giving up to date information on the Birmingham workshop.

I knew from both Quanta and the subgroup that there had been some problems over the show, and in particular over the boundaries of responsibility. My brief from the Quanta committee was to give West Midlands any support they needed in running the workshop. Little did I know that I would sound its death knell.

The West Midlands Quanta subgroup have not run a Quanta workshop before, but Mike Bedford White, its secretary and treasurer, has much organising experience. He could solve most of the problems I threw at him, but to my more workshop experienced eye two problems quickly emerged. The traders had far more goods to load and unload than Mike had envisaged. The pub venue had no car park or forecourt and on both sides of the road there were double yellow lines. We also could not offer traders the space to display their goods. The second point was one that even the most experienced show organisers have been known to miss. A check on the electrical provision indicated that a workshop at this venue would involve an irresponsible and unsafe level of daisy chaining.

My verdict was clear. This was an excellent setting for a meeting or discussion, but not for a full blown Quanta workshop.

You do not lightly stick the knife into an event that has already been advertised, but I had to do it. I arrived home two hours later than my normal bedtime. The next morning I would have to write a lengthy report to the Quanta committee. I also had to warn Jochen of the problems. And then there was the telephone problem. I had been promised it would be repaired within 24 hours, but suppose it was not. It was a perfect recipe for a sleepless night.

As I settled down on Tuesday morning to write my report there was a welcome sign. A telephone engineer was working in the manhole outside my flat. Half an hour later my telephone was back in action and I could email Jochen with the news.

Writing a report for Quanta took longer because I could not prejudge the committee's likely reaction. I gave my advice that the venue was unsuitable, but also provided the committee with the information they needed if they decided to go ahead with the show. Early in the afternoon I emailed a 2Mb file of 1000 words plus photos.

Within an hour I received a first reaction. Within 24 hours the committee had made their decision. The easiest thing for the Quanta committee would have been to have scrapped the Midlands venue and move the show elsewhere, possibly Manchester. However they did not choose the easy option as a matter of principle.

Some years ago I had researched the demography of Quanta and had identified the Midlands as being a black spot where there was a concentration of members, but no workshops. The Quanta committee had decided to hold a show in this area as a matter of policy and they were not going go back on this decision. They would find an alternative venue.

On Wednesday I had to forget all about Birmingham as a private matter intervened. In the last few months I have doing battle with Royal Mail and a government department. Royal Mail had performed badly. I had paid extra postage on seven letters for a proof of delivery, but they could provide this in just one case. They were trying to avoid paying full compensation, and a formal complaint to the official consumer body had produced no change in their attitude. But then out of the blue came a formal apology and a cheque to compensate me after a national newspaper had taken up the case.

I had to drop everything and write to the reporter thanking her, send her up to date information and documentation and give written permission for the material to be used in the paper.

On Thursday evening it was back to workshop action, with the telephone lines between Manchester, Birmingham and Derby glowing red hot as we put together a solution to the problem. At one stage Mike Bedford White went out to instantly check a hall. It was highly suitable, but unfortunately was a private club whose terms did not allow hire to third parties. In the end we chose the Holiday Inn in Solihull, a venue Sarah Gilpin had been successful in securing.

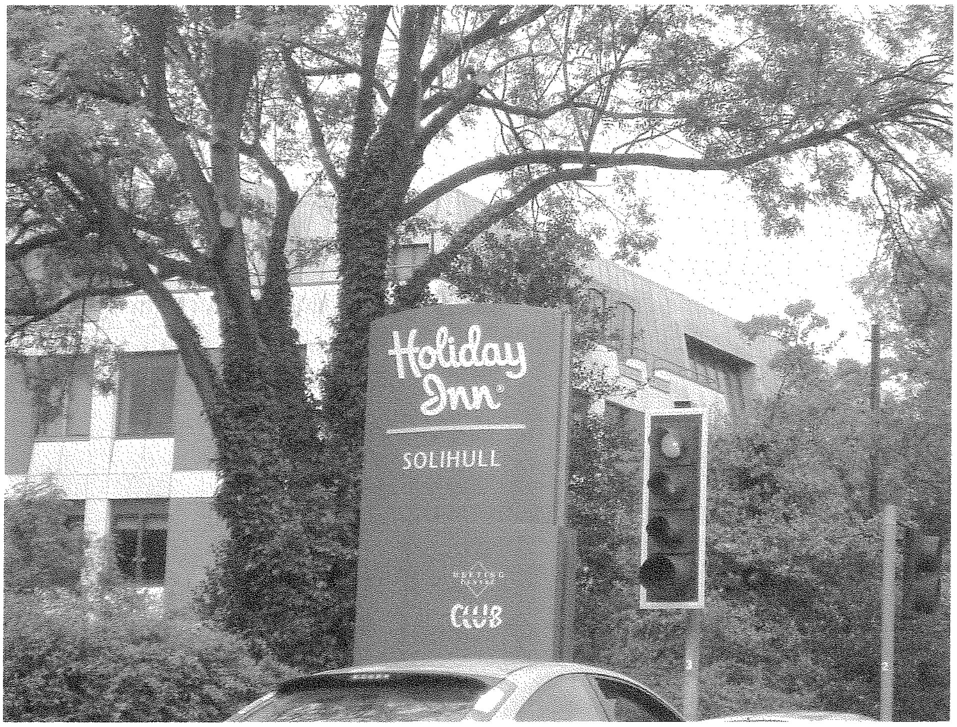
This show will be in a more luxurious setting than the average Quanta workshop, but the committee had a good reason for wanting to keep the show in the Midlands. Off the record QL Today understands they want to use that weekend to look at other ways to give Quanta a better presence in the region.

Given the effort the committee have made, we hope that members and non-members will now make an equivalent effort to attend the show. Two people have already been pencilled in as speakers and we are hoping for a third. Without giving away too many secrets you should be prepared to be challenged by a non-SMSQ-E non-Microsoft dependent vision of the QL.

As the show is in a hotel there should be no parking problems and road access is easy as junction 5 of the M42 is on the outskirts of Solihull.

For travellers by public transport there is a disappointment. Solihull station will be closed for engineering work on 6th October. There will be a replacement bus service from Birmingham Snow Hill station but this will not stop at Birmingham Moor Street. Buses depart at 20 minute intervals, but you should allow up to an extra 30 minutes travelling time.

The hotel is easy to find from the bus and rail station. (Your QL Today reporter had no difficulty in spite of forgetting his Birmingham map!) Just go to the end of the station approach road and



cross over into Homer Road. Then follow the signs to the police station or the magistrates court. The Holiday Inn is a little further up the road on the right hand side.

Finally for non-QL-er partners Solihull has a pleasant flower-filled shopping centre 5 - 10 minutes away from the hotel. The more adventurous, given the public transport problems, could also visit Birmingham. This has a high reputation as a shopping centre following the re-development of the Bull Ring together with an extensive and fascinating market area.

My week ended in a much better way than it began. The big remaining problem was how to spend the Royal Mail cheque. No such luck! They had made it out incorrectly and my bank refused to accept it. Aargh!

Contacting the Authors

As promised by our Editor, here is a list of e-mail adresses of the authors who are happy to have them published. Some authors explicitly do not want their e-mail address to be published. Of course, we respect their wish. For authors not listed here, QL Today will try to forward inquiries, provided, we know the current e-mail address. Not everybody who has ever written in QL Today has replied yet, so the list is not complete.

Geoff Wicks	gwicks@beeb.net
Marcel Kilgus	marcel@kilgus.net
Simon N Goodwin	simon@studio.woden.com
George Gwilt	gdgqler@gmail.com
Per Witte	pj@witte.fsbusiness.co.uk
Dilwyn Jones	dilwyn@uk6.net
Herb Schaaf	hlschaaf@UDel.Edu
Norman Dunbar	NormanDunbar@users.sourceforge.net
Stephen Poole	Stevepoole@minitel.net
Roy Wood	roywood@qbranch.demon.co.uk
Jochen Merz	smsq@j-m-s.com
Jim Hunkins	contact through his website: http://jdh-stech.com/#WhoWeAre
David Denham	contact through QL Today
Ian Pine	contact through QL Today
Gerard Phelan	contact through QL Today

We will try to provide you with an updated (and hopefully longer) list from time to time.

Byts of Wood

by Roy Wood

So here we are at the start of another volume and, gratifyingly, we have lost very few readers. While this is a good situation I can't help but wonder what it is that keeps us all messing around with the QL after all these years. There is a certain aspect of this which is understandable. We are all familiar with the way in which the operating system works and many of us are more or less conversant with the various programming languages the QL supports. In this regard it is a good system to mess around with but many of the QL luminaries are also programming experts in their own right on more mainstream systems so it is not like they are the sort of person who has learned one system and sticks to it because they cannot - or will not - make the effort to learn another. In spite of this, though, there is a a great lack of new programs emerging onto the scene.

As someone who operates as a trader this is a source of some concern to me. Most of the programs I list for sale are now over three years old and that is just the time of the last update. The actual genesis of the program often goes back much further.

Jonathan Hudson, who was capable of displaying quite a spikey persona when riled (and it was not that hard to do that) ported many good programs over to the QL and wrote a few from scratch too. Right now we do not seem to have anyone fulfilling that capacity on the QL scene. I suppose it could be that there are fewer smaller programs out there which can be ported over and larger ones could not be accommodated by the native QL's memory - even at its fully Super Gold Carded expansion. But every now and then someone pops up on the Users Internet list and suggests that so and so could be ported over to be used on a QL and that this would be 'a trivial matter'. None of it ever appears, however, and these people talk a good game but don't often manage to stay at the crease long enough to score.

TCP/IP

While not in the same league the much trumpeted TCP/IP stack that was announced back in the QL 2000 days is still not with us and that would be a very useful thing to have available. There are some QL systems that have a functioning TCP/IP stack. QPC2 for instance had one until Vista came along and broke it (it still works on XP/2000) and the Q40/Q60 running QDOS Classic will also do it - I gather, but few other systems can use TCP/IP communications. Of course the major setback for a native QL system would also be the lack of any kind of ethernet port and that takes us into hardware territory.

While we are talking about TCP/IP (and I have struggled with my better nature and 'sleeping dogs' principles about whether or not to mention this) I have been wondering if anyone had noticed the exchange on the Users list between Wolfgang Lenerz and Peter Graf. Peter announced he had a working TCP/IP stack but while SMSQ/E was not completely free he could not release it. When pressed what difference this made he said it required changes to be made to SMSQ/E before it would work. Wolfgang, reasonable man that he is, offered to make changes to the system if it would lead to the release of a working stack. All Peter had to do was to give him the details of the parts that needed changing and it would be done, There was no response. I leave you, my readers, to decide why that should be.

New Designs From the Past

There is precious little excitement in the QL-sphere these days so it was a bit of an unusual thing when Malcom Cadman put up the designs that the Sinclair hardware designer, Rick Dickinson, did for the QL successor. For a while the user list positively buzzed with excited voices. It was like a kids bedroom on Christmas morning except that none of these things were real or available. Easy to see why it cause such a frisson of interest, though. The designs, which you have probably already seen elsewhere in this magazine, are really stunning. Of course they prompted the usual calls of, 'can we still make them?', 'Who can get these made?' etc. Fascinating to see that some people really have such enthusiasm for QL matters still.

It also amazes me that, considering the wealth of talent Sir Clive had at his disposal at the time, he really made such a hash of things. Designs such as these would have beaten the staid designs Amstrad was touting around at the time and may even have given Apple a run for its money. Sir Clive, however, showing his usual business accumen, decided to concentrate on recycling microdrive motors as engines for bicycles and producing a range of odd vehicles that no one wanted to buy - or even be seen riding in or on.

Such is the Great British eccentric ethos, I suppose. Why take a great idea and make something of it when you can be completely off the wall?

Funny that this should come to light just now though, with the QL scene in one of its regular doldrums. When you look at the pace of development over the years you can see it is never a smooth curve but rather proceeds in small jerks with one development setting off another. New QL based hardware would, however seem to be a rather far off prospect.

A Wider Malaise?

The mainstream computing industry would also seem to be in a fit of the doldrums. Reading some of the reports that have come out over the last couple of months it would seem that sales are, in general, a lot lower than the same period last year. The summer is a traditional back water for computer sales because most of these sales are consumer based rather than business oriented. It would seem that development has reached a plateau. Not because it has nowhere to go but because it cannot see that point. This is interesting because it reflects the position of the QL a few years back.

Computing had traditionally had three main axies of use. One strand we have the business use which is concerned with word processing and database manipulation. The office priorities were large and reliable storage, the ability of juggle large amounts of facts and figures and the production of documents and reports (lets not mention death by Power Point at this stage I will only get distressed). Most of these needs were filled quite adequately by the generation of machines that came out up to five years ago. Hard drives got bigger and Raid Arrays more complex but the basic office computer would often have a 1GHz Celeron Socket 370 CPU because, coupled with a large hard drive and a 100M network you could do most office work on this with a degree of speed.

Consequentially most offices and home office uses just stayed with these machines. Often running Windows 2000 and the Office products linked with that because they were among the first reliable O/S and applications that Microsoft produced. I know I will get some stick for this but it is basically true. Five years ago Linux was an unfriendly and difficult solutions and the free office versions were flaky, not the least for their handling of the M$ standard file formats.

The computing industry pinned its hopes on the home users. This seemed a better bet for raking in the megabucks. Gamers wanted fast graphics and high speed displays. Each new game that came out used a more sophisticated rendering engine which, in turn, needed a more powerful platform to run it. Then along came the Playstaion. Why slug your games machine by making it run an O/S with a vast overhang of processes that your games were never going to use? Ratchet up the graphics and get the CPU to do nothing except play the game. That way you have a very efficient games machine but, crucially, the flow of users constantly looking for the next PC hardware upgrade began to dry out. Few people now play games on a PC as can been seen by looking at the games for sale in your local stores. They are nearly all for games platforms and very few for a PC.

Of course the computer business movers and shakers kept plugging away at that 'Home Media Centre' concept but I have seen few of these in people's homes. Most cable and satellite providers allow you to store or time shift programs and you can have a comprehensive setup of discrete boxes to create DVDs from TV programs, play high definition audio and watch TV without the handicap of a M$ operating system getting in the way.

This leaves us with the graphics boys. Those guys in the darkened rooms producing animations and high quality images for TV. Film and the advertising industry. In some ways these are the only currency left to the computer hardware manufacturers. and while they have a need for speed it can only be a matter of time before their demands flatten out.

Software Driven?

I have mentioned before that hardware and software have a symbiotic relationship. better hardware will open doors for programmers and better programs make more demands on the hardware so it responds by upping the specs. I have also mentioned before that this is, in my opinion, the first wrong turning we made in the QL community. Programmers should have written software that would only run on the higher end systems, needed more RAM, better graphics etc. This is not a greed for more sales speaking here but a recognition that advances need to be driven and that, when consumers stop saying, 'I want a machine that can run this piece of software' or 'I want software to do this task but the hardware won't handle it', The algae starts to slit up the pond and higher end lifeforms migrate to more refined waters,

When we should have had the courage to release software that would not run on a black box with a Gold Card in it we baulked at the task and slugged the software. Admittedly we had some idiocentric programmers. People who eschewed the use of a mouse or anything higher than the Mode 8 colour scheme but that was only part of the story. Back in the real world (PC World that is) Bill Gates latest flagship O/S has been pretty much universally ignored. Sales of the operating system on disk are low and many of the machines that sell supplied with it also have 'downgrade vouchers' which allow users to turn them back into Windows XP. This is somewhat of the reverse of what I was saying above but it makes sense when you consider that many of the changes in Vista are pure cosmetics and, what advantages there are in the new system, are not obvious to Joe Punter who has not really mastered the old XP O/S and is constantly surprised when someone uses 'the other button' on the mouse.

In our case we did make a big leap with the release of SMSQ/E. It was a vast improvement on the original O/S and was so much more of a stable platform because it stayed the same if you were running it on a QXL, QPC, Atari or QL. Maybe the colours were too long in coming and maybe the hardware did not quite keep pace with the aspirations of some of the people active on the QL scene 6-10 years or so ago but designing and making hardware for such a small user base was never going to be a financially rewarding task and it is not surprising that the major players fell by the wayside. It was a shame that the rivalry between Stuart Honeyball and Ron Dunnett did not drive them into developing the next generation of Gold Cards because, at that point, it was still viable and would have kicked the systems up a notch.

The Power of History

The QL has had a long history for a system which was written off at its launch and it remained, for a long time, a system which did things in a different way. It was economical on memory and had an ease of access through SuperBasic which many other platforms lacked. I am not writing this off right now. I am not saying 'Well it has been good but - so long and thanks for all the fish' We could still inject some life back into the system but only by agreeing that the old black box was good in its time but now the system can be run on many platforms and we should make the most of this. But we do need to make a little effort.

A Hardwaring Giant

One name that has been absent from the QL Users Group for a while has been Nasta but he re-appeared the other week when Romaldo Parodi asked what happened to his web site. It seems that site was hosted by Spodmail which has gone under. He said he still had a copy of the site on his computer and he would put it back up when he had found another host.

Someone also asked him about schematics and other data for the Qubide. This, he says, is still available in the file section of the Ql hardware yahoo group at

http://groups.yahoo.com/group/QLhardware/

I realise that this exchange appeared on the QL Users list but I thought I would pass it on here for anyone who missed it. Also because it fitted in well with the discussions on hardware and QL systems in general. Nasta designed a sizable proportion of the more modern devices in use on native QL systems and, of course, designed the Aurora Motherboard.

There were a lot of noises about new Aurora boards and Super Gold Card replacements a couple of years back it would be interesting to know if any of this is still being considered. Given our limited numbers, though, it is probably not financially viable in terms of small scale fabrication. Which is a shame.

Endpiece

Much of the above is only QL related in a general way and it is mostly my thoughts on the computer industry in general. In many ways it reflects the QL's path on a macro scale. At the moment it is still in the 1988 state that the QL was in. Many of the smaller PC shops have closed down, many of the smaller manufacturers and software houses have fallen by the wayside and hardware prices have dropped. The general public - those that Stuart Honeyball would always refer to as 'The Great Unwashed' - was only interested in computers as a means to an end - playing games, downloading music or porn, stuff like that. Only a fraction understand how it works and only a fraction of these can put together a line of code. Slowly the computer is beginning to metamorph itself and integrate itself into everyday life in a form which is barely recognisable as the computer we know. Its functions are being subsumed into everyday consumer electronics. In the same way that people now rarely delve into the engines of their cars people rarely undo the screws on the back of the tower - in fact most sales these days, are of laptops which don't come apart easily and don't go back together once they have. Tinkerers are a dying breed. Lets hope we can hold out for a while longer.

The QL Show Agenda

Quanta Autumn Workshop in Birmingham New venue!

Saturday, 6th of October, 10:00 to 16:00
The Holiday Inn, 61 Homer Road, Solihull, Birmingham
On-site Parking. Non-Quanta members welcome!

Travel Instructions:
By Road: Due to extensive road works in the Birmingham area it is recommended that whichever direction you are travelling from, you navigate to Junction 5 of the M42 Motorway.
By Rail: Solihull Railway Station is less than half a mile from the venue.
By Air: Birmingham International Airport is less than three miles from the venue.

Several "famous" visitors are expected to come to the show, and various talks, demonastrations etc. are being planned.

QL Meeting in Eindhoven

Saturday, 20th of October, 10:00 to 16:00
Pleincollege St. Joris, Roostenlaan 296

Thanks to the organiser, Sjef van de Molengraaf, the meetings at Eindhoven continue. Same venue as always J-M-S will be there, as always.
At the time we prepare and print this issue, it is not clear if Q Branch and TF Services can make it. Please check the Q Branch website, J-M-S website or QL Today's website.

The Next Issue

We plan to have the next issue ready for you shortly before Christmas and hope that it will reach you in time. As always, it depends on how quickly we get reviews, articles etc.
We need more material, as always. The more we get and the sooner we get it, the quicker the next issue will be in your hands, and the better it will be. Hope to meet you at one of the forthcoming QL shows - your QL Today Team!